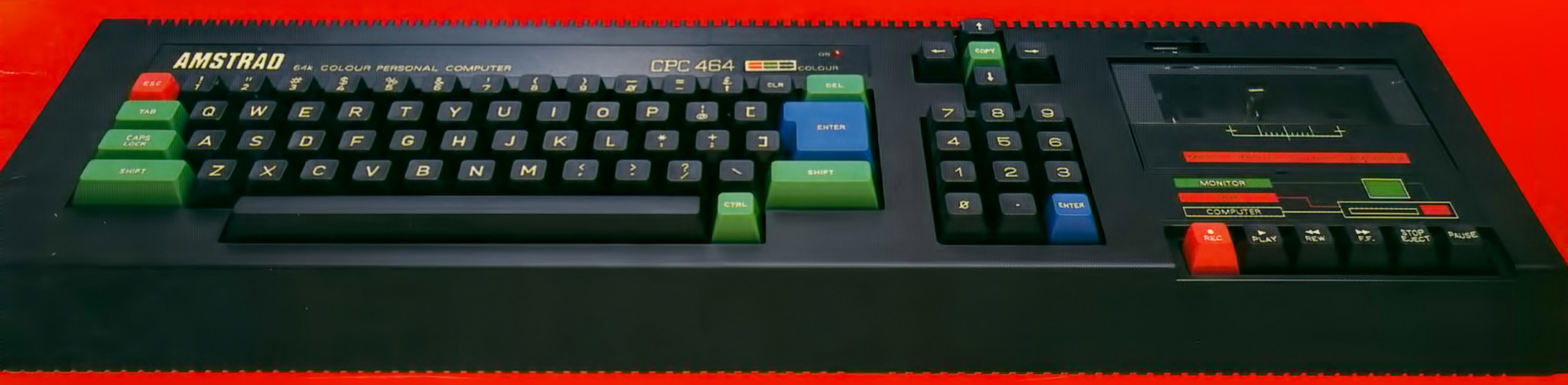
AMSTRAD

CPC464

TOPTRONICS

KÄYTTÄJÄN OPAS



CPC 464 HENKILÖKOHTAINEN 64K VÄRITIETOKONE

KÄSIKIRJAN KÄYTTÄJÄLLE

Tietokoneet ovat kehittyneet melko pitkälle varsin lyhyessä ajassa. Voidaan todeta, että tietokonetekniikan kehitys on varmasti kaikkein hämmästyttävintä verrattaessa sitä muiden tekniikan alojen kehitykseen tällä vuosisadalla.

Tietokonelaitteistojen ja -ohjelmistojen ominaisuudet ovat kehittyneet niin nopeasti, etteivät edes nykyiset käyttäjät tahdo pysyä kehityksessä mukana. Niinpä tarvittaisiinkin useita tuhansia sivuja kuvaamaan CPC 464-laitteiston ja sen käyttöjärjestelmän ominaisuuksia sekä koneen BASICin hienouksia ja käyttömahdollisuuksia.

Siksi tämä käsikirja onkin suppea johdanto CPC 464:n ja sen ohjelmiston suomiin mahdollisuuksiin. Käsikirjan seuraksi tullaan myöhemmin julkaisemaan useita tarkempia ja yksityiskohtaisempia ohjekirjoja.

Jos olet aiemmin ollut tekemisissä jonkun toisen BASIC-kielen murteen kanssa, käy tutustuminen AMSTRADin BASIC-kielen runkoon varsin nopeasti. Myös aloittelijat huomaavat sangen pian arvostavansa tässä koneessa käytettävän kielen selkeää ja yksiselitteistä terminologiaa. AMSTRADin BASIC-kieltä laadittaessa on erityisesti haluttu välttää kummallisuuksia, joita esiintyy useissa vakiintumattomissa BASICin tulkinnoissa. Samoin kieleen on haluttu lisätä joukko tosiaikaisia (eli reaaliaikaisia) perusominaisuuksia, joita ei aiemmin ole ollut saatavana halvempiin tietokoneisiin.

Käsikirja on jaettu kolmeen osaan

Käsikirjan alussa on aloittelijoille tarkoitettu peruskurssi. Se on laadittu erityisesti niille, joilla ei ole aikaisempaa kokemusta tietokoneiden parissa. Tässä käsikirjan osassa selvitetään tietokoneisiin liittyviä käsitteitä ja terminologiaa (eli sanoja ja sanontoja). Suosittelemme, että käyt läpi tämän peruskurssin, ellet ole aikaisemmin käyttänyt tietokonetta siinä määrin, että osaat tehdä itse pieniä ohjelmia.

Jos sinulla on jo aikaisempaa kokemusta mikrotietokoneista, voit aloittaa suoraan luvusta 1. Luvussa 1 käsitellään uudelleen eräitä tärkeitä seikkoja, jotka koskevat laitteiston käyttökuntoon kytkemistä ja laitteistoon tutustumista. Luvussa 1 on kuitenkin keskitytty enemmän CPC 464-järjestelmän erikoisominaisuuksien käsittelyyn. Samalla on oletettu, että luvun 1 lukija on jo jossain määrin perillä käytettävästä terminologiasta.

Käyttäjää johdatellaan laajalti CPC 464:n moniin kiehtoviin ominaisuuksiin kaikissa käsikirjan toiseen osaan kuuluvissa luvuissa, joissa käsitellään alkeita. Eräitä perusasioita toistetaan useampaan kertaan toisaalta siksi, että niitä on haluttu korostaa, ja toisaalta siksi, että monet lukijat haluavat varmaan paneutua suoraan ääni- ja grafiikkamahdollisuuksiin saatuaan mahdollisimman lyhyen opastuksen näppäimistön toimintoihin ja opittuaan järjestelmällisesti BASICin oleellisimmat käskyt.

AMSTRADin BASIC-opas (Guide to BASIC) on opetuskurssi, jossa CPC 464:n rajattomia mahdollisuuksia yhdistettynä opetus-, peli- ja "oikeana" tietokoneena käsitellään laajalti ja läpikotaisin, jotta CPC 464:n käyttäjä oivaltaisi kaikki koneen suomat mahdollisuudet. Jos siis haluat oppia asiat perusteellisesti, suosittelemme sinulle tämän BASIC-oppaan hankintaa - ellet jo ole hankkinut sitä!

Tämän käsikirjan lopussa on runsaasti liitteitä, joissa käsitellään varsin laajalti tietokoneisiin liittyviä käsitteitä sekä CPC 464:n konekohtaisia seikkoja.

Toivotammekin näin lopuksi sinulle onnea ja menestystä CPC 464-tietokoneen parissa - olisit tuskin voinut sijoittaa rahojasi viisaammin hankkiessasi CPC 464-tietokoneen, jonka suomat mahdollisuudet oppia ymmärtämään kaikki tietokoneisiin liittyvät asiat ovat sangen laajat. Paras tapa tutustua tietokoneisiin ja niiden ohjelmiin on käyttää tietokonetta - ja CPC 464 on erityisen "käyttäjäystävällinen".

AMSOFT on osa AMSTRAD Consumer Electronics plc-yritystä.

Tässä käsikirjassa mainitut ohjelmat ja laitteet on valmistettu vaivoja säästämättä. Aina on kuitenkin olemassa myös virheiden esiintymisen mahdollisuus. AMSOFT ja AMSTRAD eivät ole vastuussa mahdollisista vahingoista, jotka voivat aiheutua tässä käsikirjassa mainituissa laitteissa tai ohjelmissa esiintyvistä virheistä, tai laitteiden tai ohjelmien huolimattomasta tai virheellisestä käytöstä.

Kaikki korjaus- ja huoltotoimenpiteet on parasta antaa ammattitaitoisen huoltomekaanikon suoritettaviksi.

AMSOFT ottaa kernaasti vastaan tätä käsikirjaa tai käsikirjassa mainittuja ohjelmia tai tuotteita koskevia kommentteja ja ehdotuksia.
Kirjeet tulee lähettää osoitteella:

AMSOFT, 169 Kings Road, Brentwood, Essex CM14 4EF, ENGLAND

Z80 tarkoittaa tässä käsikirjassa Zilog Inc.´in prosessoria.

AMSTRAD on AMSTRAD Consumer Electronics plc:n rekisteröity tavaramerkki. Sanan AMSTRAD tai tavaramerkin luvaton käyttö on ehdottomasti kielletty.

Maahantuoja: Toptronics Ky, Nuppulantie 35, 20310 TURKU, puh. (921) 546 666.

Suomennos: Jukka Huhtanen

TÄRKEÄÄ!

Pidä mielessä lukiessasi tätä käsikirjaa, että tekstissä on käytetty erilaisia kirjoitustapoja haluttaessa viitata ohjelmiin, tai tietokoneen [NÄPPÄIMIIN] jotka eivät aiheuta merkin tulostumista ruutuun, tai <yleisiin selityksiin>, jotka liittyvät ohjelmointisanoihin mutta joita ei syötetä näppäimistöltä osana tietokoneelle annettavaa käskyä.

1. Laitteiston saa kytkeä sähköverkkoon vain luvussa "Kytkemme tietokonelaitteiston käyttökuntoon" olevien ohjeiden mukaisesti.

2. Älä milloinkaan kytke tietokoneen näppäimistöä, monitoria tai virtalähde/modulaattoriyksikköä mihinkään muuhun laitteeseen tai virtalähteeseen kuin mitä tässä käsikirjassa on neuvottu. Tämän neuvon noudattamatta jättäminen aiheuttaa takuun raukeamisen, ja siitä voi olla seurauksena laitteiston vioittuminen pahemman kerran.

3. Pidä kukkamaljakot, juomalasit, yms. riittävän kaukana tietokoneen näppäimistöstä, monitorista ja virtalähde/modulaattoriyksiköstä. Jos nestettä pääsee läikkymään johonkin edellä mainituista yksiköistä, saattaa se aiheuttaa vakavia vaurioita. Tällaisen vahingon sattuessa on syytä ottaa yhteys laitteen myyjään tai ammattitaitoiseen huoltoliikkeeseen.

4. Älä peitä tietokoneen näppäimistön, monitorin tai virtalähde/modulaattoriyksikön yläosassa tai takaosassa olevia ilmanvaihtoaukkoja.

5. Kaikki CPC 464:n muistiin tallennetut tiedot häviävät, kun virta katkaistaan laitteistosta. Jos haluat tallentaa ohjelman kasetille, lue Luku 2 sen jälkeen kun olet ensin lukenut läpi "PERUSTEET".

6. Laitteistoon kuuluvassa kasettinauhurissa kannattaa käyttää erityisesti tietokonekäyttöön tarkoitettuja kasetteja. Kasettinauhurissa voi kuitenkin käyttää yhtä hyvin tunnettujen valmistajien hyvälaatuisia äänentoistolaitteisiin tarkoitettuja C-kasetteja edellyttäen, etteivät ne ole CrO_2- tai "metalli"-nauhoja eivätkä ne ole kestoltaan pitempiä kuin 90 minuuttia (C-90).

   Jotta kasetille talletettujen ohjelmien löytäminen olisi helpompaa, kannattaa kasettinauhurissa käyttää C-12 -kasetteja (6 minuuttia molemmilla puolilla).

7. CPC 464-tietokoneeseen ei voi ladata eikä siinä voi ajaa muunmerkkisten tietokoneiden ohjelmakasetteja.

8. Laitteiston kasettinauhurin äänityspainike (REC) ei painu alas, jos käyttämäsi kasetin takareunasta on poistettu pieni muovikieleke. Näiden varmuuskielekkeiden tehtävänä on juuri estää äänitys, kun kyseisen kasetin puolta vastaava muovikieleke on poistettu. Äänityspainiketta ei saa tällaisessa tapauksessa painaa väkisin alas, sillä muuten mekanismi vioittuu. Jos haluat jälkeenpäin äänittää ohjelmia sellaiselle kasetille josta muovikieleke (tai molemmat kielekkeet) on poistettu, peitä aukko (tai aukot) pienellä teipin palalla. Tämän jälkeen äänittäminen onnistuu normaaliin tapaan.

9. Muista varmistaa, että kasetilla oleva nauha on kelautunut ensimmäisen osan (muovisen johtonauhan) ohi, ennen kuin alat tallettaa ohjelmaa nauhalle.

10. Pidä erityisen hyvä huoli siitä, ettet käytä etkä säilytä CPC 464-järjestelmän laitteita suorassa auringonvalossa tai paikoissa, joissa on hyvin kuumaa, kylmää, kosteaa tai pölyistä. Suojaa laitteet myös iskuilta ja tärinöiltä. Paina mieleesi, ettei ohjelmakasetteja saa koskaan säilyttää minkäänlaisten magneettikenttien läheisyydessä. Haitallisia magneettikenttiä esiintyy esim. kaiuttimien tai suurten sähkömoottoreiden läheisyydessä.

11. Säilytä kasetteja aina niiden omissa säilytyskoteloissa. Kun pidät kaseteista muutenkin hyvää huolta ja puhdistat kasettinauhurin mekanismit säännöllisin välein, pitäisi ohjelmien tallettaminen nauhalle ja ohjelmien lataaminen nauhalta tietokoneen muistiin sujua ongelmitta.

12. Laitteiden sisällä ei ole sellaisia osia, joita käyttäjä itse voisi huoltaa. Älä yritä purkaa laitteiden koteloa, äläkä yritä kajota laitteiden sisällä oleviin osiin. Käänny ammattitaitoisen huoltoliikkeen puoleen, jos laitteet tarvitsevat huoltoa.

13. Tässä käsikirjassa olevia tietoja sen enempää laitteiden kuin ohjelmienkaan osalta ei saa edes osittain jäljentää, lainata tai sovittaa muihin tarkoituksiin ilman AMSTRAD Consumer Electronics plc:n ("AMSTRADin") kirjallista lupaa.

SISÄLTÖ Sivu

KÄSIKIRJAN KÄYTTÄJÄLLE....................i

PERUSKURSSI ALOITTELIJOILLE
Helppotajuinen johdatus tietokonemaailmaan niille joilla ei ole aikaisempaa kokemusta
P1 Kytkemme tietokonelaitteiston käyttökuntoon....................1
P2 Tutustumme näppäimistöön....................10
P3 Grafiikka, äänet ja näyttömuodot....................25

1 JOHDANTO....................45
Tietokonelaitteiston kytkennät
Laitteiston käynnistys
Perustiedot näppäimistöstä
Merkkivalikoiman tulostus näyttöön
Käyttäjän määriteltävissä olevat näppäimet
Korjailu eli editointi näytössä

2 TIETOKONEEN OMA KASETTINAUHURI....................64
Ohjelmien tallettaminen muistista kasetille ja lataaminen kasetilta muistiin
Esittelykasetti
Kaksi tiedonsiirtonopeutta

3 BASICIN ALKEET....................76
Johdatus CPC 464:n BASICin periaatteisiin
AMSTRADin BASICin muotosäännöt eli syntaksi
Yksinkertaisia BASIC-harjoituksia
Tulostus PRINT-käskyllä ja näytön muotoilu
Vyöhykkeet (ZONE)

4 MUUTTUJAT, OPERAATTORIT JA TIEDOT....................84
Ehdolliset ja loogiset lausekkeet
Kohdistimen sijoitus LOCATE-käskyllä
Merkkijonot
IF - THEN -vertailut
Taulukot (array)
Tiedot (data)

5 GRAFIIKAN ALKEET....................108
AMSTRAD CPC 464:n värigrafiikan periaatteet
Skandinaaviset merkit å, ä, ö, Å, Ä ja Ö
INK-, PEN- ja PAPER-käskyt
Näyttömuodot (MODE)
Kuva-alkiot (PIXEL)
Alkupisteet (ORIGIN)
Ikkunat (WINDOW)
Yksinkertaisia grafiikkaohjelmia

6 ÄÄNEN ALKEET....................119
Minkälaisia ääniä CPC 464 pystyy tuottamaan?
Sävelet ja voimakkuusvaipat
Äänijonot
Tehosteet

Sivu

7 KIRJOITTIMET JA SAUVAOHJAIMET....130
Sauvaohjainten käyttö
JOY-käsky
Kirjoittimet
Laitteiden liitäntä CPC 464:ään

8 LYHYT KATSAUS AMSTRADIN BASIC-KIELEEN....134
Aakkosjärjestyksessä oleva lyhyt yhteenveto CPC 464:n ohjelmoinnissa käytettävistä avainsanoista ja BASIC-kielestä

9 LISÄTIETOJA OHJELMOINNISTA....188
Tietokoneen kiinteän ohjelmiston ja muiden ohjelmien välinen sisäinen organisaatio
Keskeytykset ja niiden merkitys
Ohjausmerkit
Konekielisten alirutiinien ja lausekielisten BASIC-käskyjen välinen riippuvuus

10 KESKEYTYSMAHDOLLISUUDET....193
Tosiaikaiset ominaisuudet
AFTER, EVERY ja REMAIN

LIITTEET

I Tietokoneen ominaisuuksia....196
Tietokonesanastoa

II Bitit ja tavut - binääri- ja heksadesimaalimuoto....227

III ASCII-koodit ja merkkivalikoima....230
Merkkien määritteet ja ruudukot
Näppäimistökoodit ja laajennusmerkit

IV Kokeneelle käyttäjälle tarkoitettu katsaus koneen ominaisuuksista...245

V CPC 464:n liitännät....254

VI Ruudun rivi- ja sarakemäärät eri näyttömuodoissa....257

VII Sävellystyön apuvälineitä....260
Nuotit ja säveljaksot

VIII Varatut sanat sekä virhekoodit ja virheilmoitukset....264
BASIC-avainsanat

IX Oletusarvot....269

Hakemisto....270

AMSTRAD CPC 464

PERUSKURSSI ALOITTELIJOILLE

PERUSTEET 1: KYTKEMME TIETOKONELAITTEISTON KÄYTTÖKUNTOON

Tässä luvussa ovat ohjeet CPC 464-järjestelmän laitteiden toisiinsa kytkemistä ja käynnistämistä varten.

AMSTRAD CPC 464-tietokoneen kanssa voidaan käyttää seuraavia yksiköitä:

1.1 AMSTRAD GT 64 vihreä/musta-monitoria
1.2 AMSTRAD CTM 640 värimonitoria
1.3 AMSTRAD MP 1 modulaattori/virtalähdeyksikköä ja tavallista (UHF-alueella varustettua) väritelevisiota.

Kytke tietokonejärjestelmäsi laitteet toisiinsa vastaavassa kohdassa annettujen ohjeiden mukaisesti, jonka jälkeen voit siirtyä tutkimaan varsinaisia käyttöohjeita.

1.1 AMSTRAD GT 64 vihreä/musta-monitori

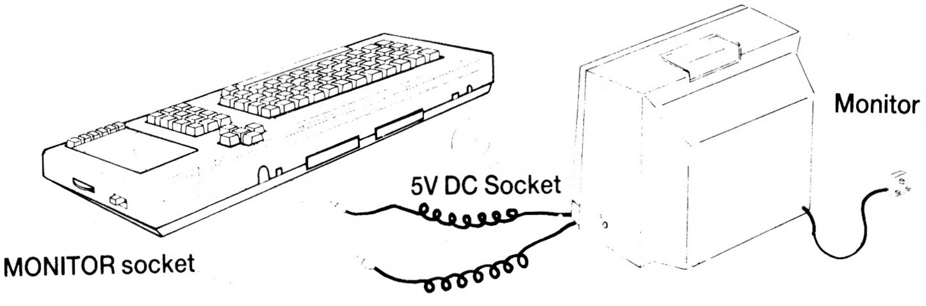
TÄRKEÄÄ!

* Laitteistossa saa käyttää vain sen mukana toimitettua verkkojohtoa ja -pistoketta.

* Irrota verkkojohto pistorasiasta aina siksi ajaksi kun laitteisto ei ole käytössä.

* Mitään sisäisiä kytkentöjä ei tarvitse suorittaa, joten älä yritä päästä käsiksi laitteiden sisällä oleviin osiin.

Tietokone sijoitetaan monitorin eteen sopivalle pöydälle pistorasian läheisyyteen. Kytke monitorista tuleva suurempi 6-napainen DIN-pistoke tietokoneen takana olevaan monitoriliitäntään (jossa on merkintä MONITOR) kuvan 1 osoittamalla tavalla. Kytke monitorista tuleva pienempi tasavirtasyötön pistoke tietokoneen takaosassa olevaan tasavirtasyötön liitäntään (jossa on merkintä 5 V DC).



Kuva 1.

Varmista, että monitorin virtakytkin (POWER) on "OFF"-asennossa (ulkona). Kytke monitorin verkkojohto tavalliseen vaihtovirtapistorasiaan (220 V AC).

Kytke nyt virta monitoriin painamalla virtakytkin sisään, ja kytke seuraavaksi virta tietokoneeseen sen oikealla puolella olevalla virtakytkimellä (POWER).

Nyt pitäisi tietokoneen näppäimistön yläpuolella keskivaiheilla olevan punaisen merkkivalon (ON) syttyä ja monitoriin pitäisi ilmestyä seuraavan näköinen kuva:

```
Amstrad 64K Microcomputer  (v1)

©1984 Amstrad Consumer Electronics plc
           and Locomotive Software Ltd.

BASIC 1.0

Ready
█
```

Kohdistin

Säädä kuvan kirkkaus sopivaksi kirkkaussäätimellä (BRIGHTNESS), etteivät silmäsi rasitu suotta kuvaa katsellessasi. Kuvan kirkkaus on sopiva, kun kuvaa on miellyttävä katsella, kuva ei häikäise ja teksti näkyy selvästi.

Säädä myös kuvan sävykkyys sävykkyyssäätimellä (CONTRAST) niin pieneksi kuin mahdollista kuvan katselumukavuuden silti kärsimättä.

GT 64-monitorissa on myös pystypidon säädin (V-HOLD). Tämä säädin tulee säätää sellaiseen asentoon, että kuva on oikeassa kohdassa keskellä kuvaruutua. Kuvassa ei myöskään saa esiintyä värinää eikä pyrkimystä "pyörimiseen".

1.2 AMSTRAD CTM 640-värimonitori

TÄRKEÄÄ!

* Laitteistossa saa käyttää vain sen mukana toimitettua verkkojohtoa ja -pistoketta.

* Irrota verkkojohto pistorasiasta aina siksi ajaksi kun laitteisto ei ole käytössä.

* Mitään sisäisiä kytkentöjä ei tarvitse suorittaa, joten älä yritä päästä käsiksi laitteiden sisällä oleviin osiin.

Tietokone sijoitetaan monitorin eteen sopivalle pöydälle pistorasian läheisyyteen. Kytke monitorista tuleva suurempi 6-napainen DIN-pistoke tietokoneen takana olevaan monitoriliitäntään (jossa on merkintä MONITOR) edellisellä sivulla olevan kuvan 1 osoittamalla tavalla. Kytke monitorista tuleva pienempi tasavirtasyötön pistoke tietokoneen takaosassa olevaan tasavirtasyötön liitäntään (jossa on merkintä 5 V DC).

Varmista, että monitorin virtakytkin (POWER) on "OFF"-asennossa (ulkona). Kytke monitorin verkkojohto tavalliseen vaihtovirtapistorasiaan (220 V AC).

Kytke nyt virta monitoriin painamalla virtakytkin sisään, ja kytke seuraavaksi virta tietokoneeseen sen oikealla puolella olevalla virtakytkimellä (POWER).

Nyt pitäisi tietokoneen näppäimistön yläpuolella keskivaiheilla olevan punaisen merkkivalon (ON) syttyä ja monitoriin pitäisi ilmestyä seuraavan näköinen kuva:

```
Amstrad 64K Microcomputer  (v1)

©1984 Amstrad Consumer Electronics plc
          and Locomotive Software Ltd.

BASIC 1.0

Ready
█
```

Kohdistin

Säädä kuvan kirkkaus sopivaksi monitorin sivulla olevalla kirkkaussäätimellä (BRIGHTNESS), etteivät silmäsi rasitu suotta kuvaa katsellessasi. Kuvan kirkkaus on sopiva, kun kuvaa on miellyttävä katsella, kuva ei häikäise ja teksti näkyy selvästi.

1.3 AMSTRAD MP1 modulaattori/virtalähdeyksikkö ja (UHF-alueella varustettu) väritelevisio

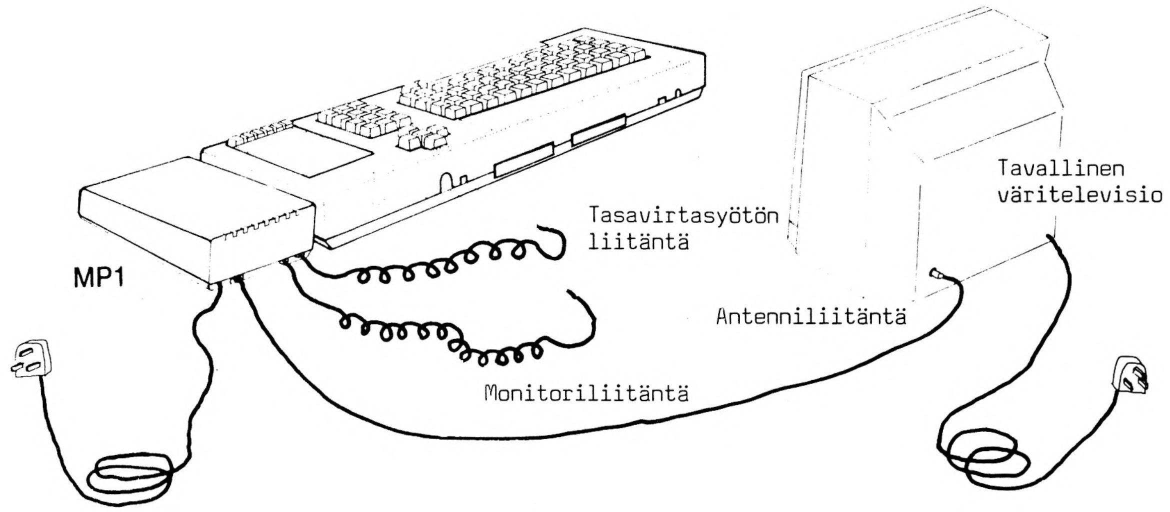
MP1 on lisäyksikkö jonka ehkä haluat hankkia, jos nykyinen laitteistosi käsittää CPC 464-tietokoneen ja GT 64 vihreä/musta-monitorin. MP1 on nimittäin laite, jonka avulla voit kytkeä tietokoneesi tavalliseen väritelevisioon ja nauttia CPC 464:n hienoista väreistä.

TÄRKEÄÄ!

* Laitteistossa saa käyttää vain sen mukana toimitettua verkkojohtoa ja -pistoketta.

* Irrota verkkojohto pistorasiasta aina siksi ajaksi kun laitteisto ei ole käytössä.

* Mitään sisäisiä kytkentöjä ei tarvitse suorittaa, joten älä yritä päästä käsiksi laitteiden sisällä oleviin osiin.



Kuva 2. MP1 modulaattori/virtalähdeyksikkö

Modulaattori/virtalähdeyksikkö (MP1) sijoitetaan tietokoneen oikealle puolelle sopivalle pöydälle television ja pistorasian läheisyyteen. Kytke MP1:stä tuleva suurempi 6-napainen DIN-pistoke tietokoneen takana olevaan monitoriliitäntään (jossa on merkintä MONITOR) kuvan 2 osoittamalla tavalla. Kytke MP1:stä tuleva pienempi tasavirtasyötön pistoke tietokoneen takaosassa olevaan tasavirtasyötön liitäntään (jossa on merkintä 5 V DC).

Kytke MP1:stä tuleva antennijohto television antenniliitäntään.

Tarkista ensin, että tietokoneen oikealla puolella oleva virtakytkin (POWER) on "OFF"-asennossa, ja kytke sen jälkeen MP1:n verkkojohto pistorasiaan.

Säädä nyt televisiosi äänenvoimakkuus aivan hiljaiselle, kytke virta televisioon ja kytke seuraavaksi virta tietokoneeseen sen oikealla puolella olevalla virtakytkimellä (POWER).

Tietokoneen näppäimistön yläpuolella keskivaiheilla olevan punaisen merkkivalon (ON) pitäisi nyt syttyä, ja seuraavaksi sinun onkin viritettävä televisiosi vastaanottamaan tietokoneen lähettämää signaalia.

Jos televisiossasi on painikkeilla toimiva kanavanvalitsin, valitse vapaa kanava painamalla kanavanvalintapainiketta. Säädä tarvittaessa kanavan hienovirityssäädintä television käyttöohjeessa kuvatulla tavalla. Tietokoneen lähettämä signaali löytyy UHF-kanavan 36 paikkeilta. Voit tarkistaa tämän seikan, jos televisiosi kanavanvalitsimessa on numeromerkinnöillä varustettu viritysasteikko. Säädä hienoviritystä tarpeen mukaan, kunnes television kuvaruudussa näkyy seuraavan näköinen kuva:

```
Amstrad 64K Microcomputer  (v1)

©1984 Amstrad Consumer Electronics plc
          and Locomotive Software Ltd.

BASIC 1.0

Ready
█
```

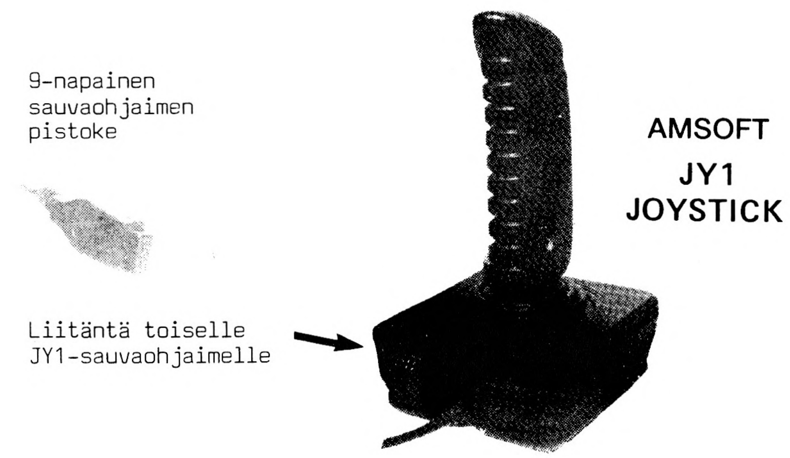
Kohdistin

Kanavan viritys on syytä suorittaa tarkasti, jotta kuva näkyy mahdollisimman terävänä. Teksti näkyy kullankeltaisena tummansinisellä pohjalla.

Jos televisiossasi on käännettävä kanavanvalitsijan nuppi, käännä nuppia kunnes kuvaruudussa näkyy edellisellä sivulla olevassa kuvassa näkyvä teksti ja kuva näkyy täysin vakaana. (Myös tässä kuva löytyy UHF-kanavan 36 paikkeilta).

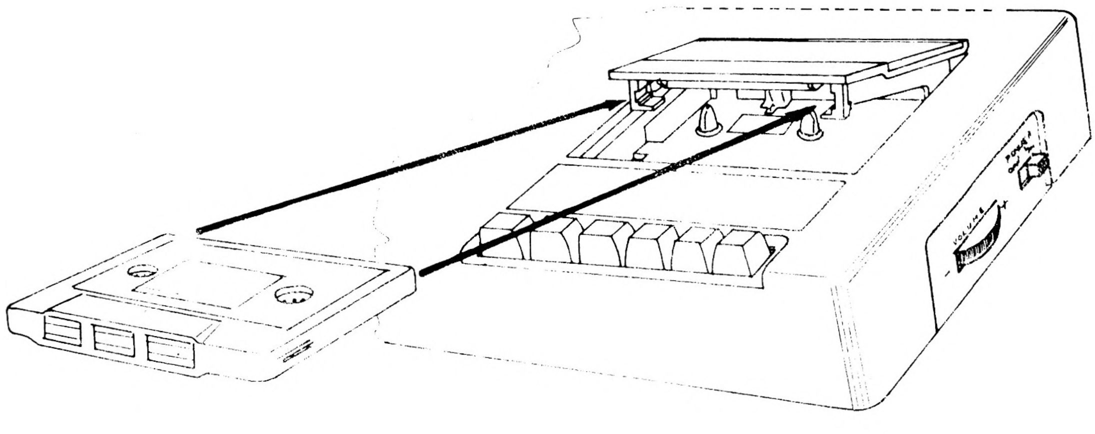
1.4 Sauvaohjain (JOYSTIC)

AMSOFTin sauvaohjain JY1 on lisävaruste jonka ehkä haluat hankkia, jos käytät CPC 464-tietokoneessa peliohjelmia joissa on mahdollisuus sauvaohjaimen käyttöön ja joissa pelin luonteeseen kuuluu tulittaminen (fire). Sauvaohjain JY1 kytketään tietokoneen takaosassa olevaan 9-napaiseen käyttäjänporttiliitäntään (USER PORTS I/O). AMSTRAD CPC 464-tietokoneessa voidaan käyttää kahta sauvaohjainta samanaikaisesti. Toinen JY1-sauvaohjain kytketään ensimmäisessä sauvaohjaimessa olevaan liitäntään.



1.5 Esittelykasetti

Olet jo ehkä löytänytkin esittelykasetin, joka on pakattu tietokoneen pakkauksen toiseen polystyreeni-päätysuojukseen. Avaa tietokoneen kasettinauhurin kansi painamalla pysäytys/kasetinpoistopainiketta [STOP/EJECT], ja laita sen jälkeen kasetti kasettipesään kuvan 3 osoittamalla tavalla - varmista, että 1-puoli (SIDE 1) tulee ylöspäin.



Kuva 3. Kasetin oikea laittotapa kasettinauhuriin

Sulje kasettipesän kansi niin että se lukittuu. Paina seuraavaksi kasettinauhurin takaisinkelauspainiketta [REW] ja varmista, että nauha on kelautunut aivan alkuun. Paina pysäytys/kasetinpoistopainiketta [STOP/EJECT] heti kun nauha pysähtyy. Nollaa nauhalaskijan lukemaksi 000 painamalla nauhalaskijan nollauspainiketta (RESET).

Paina [CTRL]-näppäin alas ja paina SAMANAIKAISESTI pientä [ENTER]-näppäintä, joka on kasettinauhurin vieressä olevan numeronäppäimistön oikeassa alakulmassa. Tällöin kuvaruutuun ilmestyy seuraava teksti:

```
RUN"
Press PLAY then any key:
```

Paina nyt kasettinauhurin etureunassa olevaa [PLAY]-painiketta, kunnes se lukittuu tukevasti ala-asentoon. Paina seuraavaksi mitä hyvänsä kirjain- tai numeronäppäintä tai jompaa kumpaa [ENTER]-näppäintä - tai näppäimistön alareunassa olevaa pitkää välilyöntinäppäintä.

Tällöin nauha lähtee liikkeelle ja kuvaruutuun ilmestyy seuraava teksti lyhyen ajan kuluttua:

```
Loading WELCOME 1 block 1
```

Nauhan lataaminen kestää noin viisi minuuttia, ja voit seurata lataamisen edistymistä kuvaruudulla lohkonumeroiden (block) vaihtuessa 1...2...3...jne., kunnes nauha pysähtyy. Esittelyohjelma käynnistyy sillä hetkellä kun nauha pysähtyy. Sinun ei tarvitse tehdä muuta kuin otat mukavan asennon ja katselet mitä ruudussa näkyy. Tietokone ajaa esittelyohjelmaa jatkuvasti kerta toisensa perään, joten voit painaa [ESC]-näppäintä kaksi kertaa kun et enää halua katsella esittelyohjelmaa. Ohjelma pysähtyy painaessasi [ESC]-näppäintä kahdesti, jonka jälkeen voit painaa pysäytys/kasetinpoistopainiketta [STOP/EJECT], kääntää kasetin toisin päin ja ajaa 2-puolen ohjelman edellä olevien ohjeiden mukaisesti.

Muista kuitenkin painaa takaisinkelauspainiketta [REW] käännettyäsi kasetin toisin päin kasettipesässä varmistaaksesi sen, että nauha on kelattu aivan alkuun.

Paina [CTRL]-näppäin alas ja paina SAMANAIKAISESTI pientä [ENTER]-näppäintä, joka on kasettinauhurin vieressä olevan numeronäppäimistön oikeassa alakulmassa. Tällöin kuvaruutuun ilmestyy seuraava teksti:

```
RUN"
Press PLAY then any key:
```

Paina nyt kasettinauhurin etureunassa olevaa [PLAY]-painiketta, kunnes se lukittuu tukevasti ala-asentoon. Paina seuraavaksi mitä hyvänsä kirjain- tai numeronäppäintä tai jompaa kumpaa [ENTER]-näppäintä - tai näppäimistön alareunassa olevaa pitkää välilyöntinäppäintä.

Tällöin nauha lähtee liikkeelle ja kuvaruutuun ilmestyy seuraava teksti lyhyen ajan kuluttua:

```
Loading WELCOME 2 block 1
```

Tämän jälkeen sinun tulee noudattaa kuvaruudussa näkyviä ohjeita ja näppäillä koneeseen käskyjä, joita tietokone pyytää sinua näppäilemään ohjelman edistyessä.

1.6 Muiden ohjelmakasettien lataaminen

ESITTELYKASETTI VOIDAAN LADATA JA SILLÄ OLEVAT OHJELMAT VOIDAAN AJAA VAIN EDELLISESSÄ KOHDASSA (1.5) KUVATULLA TAVALLA. Suojaamattomien BASIC-ohjelmien lataamiseen voidaan käyttää seuraavia vaihtoehtoisia tapoja. Kelaa kasettinauhuriin laittamasi kasetin nauha alkuun painamalla takaisinkelauspainiketta [REW], kunnes nauhan kelausliike pysähtyy. Nauhan pysähtyessä sinun on heti painettava pysäytys/kasetinpoistopainiketta [STOP/EJECT].

Palauta tietokone alkutilaan tyhjentämällä sen muisti. Tämä tapahtuu painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä tässä järjestyksessä pitäen kuitenkin jokaista painiketta alhaalla kunnes olet painanut viimein [ESC]-näppäintä. Tällöin ruutu tyhjenee ja siihen ilmestyy alkutekstit samalla tavalla kuin kytkettäessä virta tietokonelaitteistoon.

Seuraavissa käskyriveissä oleva [ENTER] tarkoittaa sitä, että sinun on painettava jompaa kumpaa [ENTER]-näppäintä käskyrivin päätteeksi. Älä siis näppäile näppäimistöstä sanaa ENTER! Lainausmerkki " saadaan painamalla jompaa kumpaa [SHIFT]-näppäintä samanaikaisesti näppäimistön ylärivissä olevan numeronäppäimen 2 kanssa.

Näppäile:

```
load "" [ENTER]
```

Nyt tietokone pyytää sinua painamaan [PLAY]-painiketta ja sen jälkeen mitä hyvänsä näppäintä tulostamalla ruutuun:

```
Press PLAY then any key:
```

Paina nyt kasettinauhurin etureunassa olevaa [PLAY]-painiketta, kunnes se lukittuu tukevasti ala-asentoon. Paina seuraavaksi mitä hyvänsä kirjain- tai numeronäppäintä tai jompaa kumpaa [ENTER]-näppäintä - tai näppäimistön alareunassa olevaa pitkää välilyöntinäppäintä.

Tällöin nauha lähtee liikkeelle ja kuvaruutuun ilmestyy seuraava teksti lyhyen ajan kuluttua:

```
Loading <ohjelman nimi> block 1
```

Näytössä näkyvä lohkonumero (block) kasvaa niin kauan, kunnes nauhalla oleva ohjelma on latautunut tietokoneen muistiin. Tämän jälkeen näyttöön ilmestyy seuraava viesti, joka ilmoittaa tietokoneen olevan valmiina:

```
Ready
```

Sinulla on myös toinen vaihtoehto. Voit nimittäin määritellä sen ohjelman nimen, jonka haluat latautuvan kasetilta muistiin. Tämä tapahtuu näppäilemällä:

```
load "<ohjelman nimi>" [ENTER]
```

(Muistathan, ettei merkkejä < > näppäillä ohjelman nimen alkuun ja loppuun).

Nyt tietokone pyytää sinua painamaan [PLAY]-painiketta ja sen jälkeen mitä hyvänsä näppäintä tulostamalla ruutuun:

Press PLAY then any key:

Paina nyt kasettinauhurin etureunassa olevaa [PLAY]-painiketta, kunnes se lukittuu tukevasti ala-asentoon. Paina seuraavaksi mitä hyvänsä kirjain- tai numeronäppäintä tai jompaa kumpaa [ENTER]-näppäintä - tai näppäimistön alareunassa olevaa pitkää välilyöntinäppäintä.

Tällöin nauha lähtee liikkeelle. Jos ohjelma - jonka olet pyytänyt tietokonetta lataamaan kasetilta muistiin - ei ole nauhan alkupäässä, etsii tietokone ohjelmaa nauhalta niin kauan kunnes se löytää täsmälleen sen nimisen ohjelman, jonka pyysit tietokonetta lataamaan. Ole siis huolellinen, että näppäilet ohjelman nimen oikein.

Jos tietokone löytää haluamaasi ohjelmaa nauhalta etsiessään eri nimen kuin sen jonka syötit näppäimistöltä, ilmestyy kuvaruutuun seuraavan näköinen ilmoitus:

Found <muun ohjelman nimi> block 1

Tietokone ei lataa tätä ohjelmaa muistiin vaan jatkaa etsimistään, kunnes se löytää nauhalta täsmälleen sen ohjelman nimen, jonka syötit näppäimistöltä. Tietokone ei jatka ohjelman hakua nauhalta myöskään silloin, jos painat [ESC]-näppäintä.

Kun ohjelma on löytynyt nauhalta, ilmestyy kuvaruutuun seuraava viesti:

Loading <ohjelma nimi> block 1

Lohkonumerot (block) kasvavat niin kauan, kunnes ohjelman lataaminen nauhalta muistiin päättyy. Kuvaruutuun ilmestyy seuraava viesti:

Ready

Näppäile:

run [ENTER]

Tällöin juuri lataamasi ohjelma käynnistyy. Jos tietokoneen muistissa oli jo aikaisemmin jokin ohjelma, pyyhkiytyy vanhempi ohjelma pois muistista, eikä muistiin jää muuta kuin juuri lataamasi uusi ohjelma.

Jos haluat käynnistää ohjelman ajon suoraan pyytämättä tietokonetta ensin lataamaan ohjelma nauhalta muistiin, näppäile seuraava käsky:

run "" [ENTER]

Tietokone vastaa antamaasi käskyyn seuraavasti:

Press PLAY then any key:

Paina nyt kasettinauhurin [PLAY]-painiketta ja sen jälkeen mitä hyvänsä kirjain- tai numeronäppäintä, välilyöntinäppäintä tai kumpaa hyvänsä [ENTER]-näppäintä, jolloin tietokone alkaa hakea ohjelmaa ja lataa ensimmäisen nauhalta löytämänsä ohjelman sekä käynnistää sen ajon ilman, että sinun tarvitsee antaa lisäkäskyjä näppäimistöltä. Voit pysäyttää toimenpiteiden suorituksen koska hyvänsä painamalla [ESC]-näppäintä tavalliseen tapaan.

1.7 Valmiiden (ohjelmaa sisältävien) ohjelmakasettien lataaminen

Tähän mennessä annetut ohjeet on tarkoitettu mahdollistamaan minkä hyvänsä nimisten ohjelmien lataaminen, joita CPC 464-tietokoneeseen on saatavana.

Ohjelmien oikeat latausohjeet löytyvät myös ohjelmakasettien koteloissa olevista ohjeista.

1.8 Ohjelman tallentaminen nauhalle SAVE-käskyllä

Voit tallentaa (eli äänittää) ohjelman nauhalle myöhempää käyttöä varten. Laita kasetti paikalleen kasettipesään aiemmin annettujen ohjeiden mukaisesti ja sulje kasettipesän kansi (varmista, ettei kasetin kapeasta takareunasta ole poistettu pieniä muovikielekkeitä). Kelaa nauha alkuun painamalla takaisinkelauspainiketta [REW], ja muista painaa pysäytys/kasetinpoistopainiketta [STOP/EJECT] heti kun nauha pysähtyy. Näppäile sitten seuraava käskyrivi:

save "<ohjelman nimi>" [ENTER]

Tällöin tietokone pyytää sinua painamaan kasettinauhurin äänitys- ja toistopainikkeita tulostamalla ruutuun seuraavan viestin:

Press REC and PLAY then any key:

Paina nyt kasettinauhurin [REC]- ja [PLAY]-painikkeita yhtäaikaa, kunnes ne lukittuvat tukevasti ala-asentoon. Paina seuraavaksi mitä hyvänsä näppäintä (kirjain- tai numeronäppäintä, välilyöntinäppäintä tai kumpaa hyvänsä [ENTER]-näppäintä).

Tietokone vastaa tulostamalla ruutuun seuraavan viestin:

Saving <ohjelman nimi> block 1

Kun tietokone on tallentanut ohjelman nauhalle, kasetin kelausliike pysähtyy ja ruutuun ilmestyy suoritetun tehtävän varmistukseksi sana **Ready** .

Paina nyt kasettinauhurin pysäytys/kasetinpoistopainiketta [STOP/EJECT]. Ohjelmasi on tallentunut nauhalle.

Huomaa, ettet pysty tallentamaan valmiita (myytäviä) ohjelmia ja pelejä omalle tyhjälle kasetillesi.

Tällaiset ohjelmat on suojattu luvattoman kopioinnin varalta.

PERUSTEET 2: TUTUSTUMME NÄPPÄIMISTÖÖN

Tässä luvussa käydään läpi joidenkin tietokoneen näppäinten toimintoja. Voit hypätä tämän luvun yli, jos sinulla on jo aikaisempaa kokemusta tietokoneiden käytöstä.

[ENTER]

CPC 464-tietokoneessa on kaksi [ENTER]-näppäintä. Voit syöttää näppäimistöltä näppäilemäsi tiedot tietokoneen muistiin kummalla hyvänsä näistä näppäimistä. Jokainen [ENTER]-näppäimen painallus aloittaa uuden rivin kuvaruudussa. Sinun on muistettava painaa [ENTER]-näppäintä joka kerta kun syötät näppäimistöltä jonkun käskyn tietokoneeseen.

Tästä eteenpäin tekstissä esiintyvä [ENTER] tarkoittaa sitä, että sinun on painettava [ENTER]-näppäintä jokaisen käskyn tai ohjelmarivin jälkeen.

[DEL]

Tätä näppäintä käytetään silloin, kun kuvaruudusta halutaan poistaa jokin tarpeeton merkki (esim. kirjain tai numero). Merkki poistuu kohdistimen vasemmalta puolelta.

Näppäile abcd ja näet, että kirjain d on kohdistimen vasemmalla puolella. Jos päätät, ettet tarvitse kirjainta d ja haluat poistaa sen, paina [DEL]-näppäintä yhden kerran ja huomaat, että d häviää kuvaruudusta. Jos painat vielä [DEL]-näppäintä ja pidät sen alhaalla, häviävät myös kirjaimet abc.

[SHIFT]

CPC 464-tietokoneessa on kaksi [SHIFT]-näppäintä. Jos painat jompaa kumpaa näistä näppäimistä ja pidät sen alhaalla samalla kun painat jotain kirjain- tai numeronäppäintä, ilmestyy ruutuun iso kirjain tai kyseisen näppäimen yläreunaan merkitty erikoismerkki.

Näppäile kirjain a ja pidä sen jälkeen [SHIFT]-näppäintä alhaalla ja näppäile kirjain a uudelleen. Ruutuun ilmestyy:

aA

Näppäile nyt muutamia välilyöntejä pitämällä näppäimistön alareunassa olevaa pitkää välilyöntinäppäintä alhaalla. Kokeile seuraavaksi numeronäppäinten käyttöä, jotka ovat kirjainnäppäinten yläpuolella näppäimistön ylärivissä. Näppäile numero 2 ja pidä sen jälkeen [SHIFT]-näppäintä alhaalla ja näppäile numero 2 uudelleen. Ruutuun ilmestyy:

2"

Näet nyt selvästi, mitä tapahtuu kun [SHIFT]-näppäintä pidetään alhaalla samalla kun jotain kirjain- tai numeronäppäintä painetaan. Tutustu näppäimistön eri merkkeihin painelemalla kirjain- ja numeronäppäimiä yksitellen ja sitten yhdessä [SHIFT]-näppäimen kanssa.

[CAPS LOCK]

Tällä näppäimellä on samanlainen vaikutus kuin [SHIFT]-näppäimelläkin, mutta [CAPS LOCK]-näppäintä tarvitsee painaa vain kerran. Sen jälkeen kaikki kirjaimet tulostuvat isoina kirjaimina, mutta numerot tulostuvat normaalisti. Suorita kokeilu painamalla [CAPS LOCK]-näppäintä kerran, ja näppäile sitten abcdef123456. Tällöin ruudussa näkyy:

ABCDEF123456

Huomaat, että vaikka kaikki kirjaimet ovat muuttuneet isoiksi kirjaimiksi, eivät numerot ole muuttuneet miksikään. Jos haluat näppäillä esim. jonkun numeronäppäimessä olevan erikoismerkin [CAPS LOCK]-näppäimen ollessa toiminnassa, painat vain [SHIFT]-näppäintä ennen kuin painat haluamasi merkin näppäintä, ja merkki tulostuu ruutuun. Näppäile ABCDEF123456 samalla pitäen [SHIFT]-näppäin alhaalla. Tällöin seuraavat merkit ilmestyvät ruutuun:

ABCDEF!"#$%&

Paina [CAPS LOCK]-näppäintä uudelleen, jos haluat palata kirjoittamaan pieniä kirjaimia.

Jos haluat kirjoittaa näppäimistöltä isoja kirjaimia <u>ja</u> esim. numeronäppäinten yläreunaan merkittyjä merkkejä etkä halua koko ajan pitää [SHIFT]-näppäintä painettuna, saat tämän aikaan pitämällä [CTRL]-näppäintä alhaalla ja painamalla sitten [CAPS LOCK]-näppäintä kerran. Näppäile nyt abcdef123456, jolloin ruutuun ilmestyy seuraava rivi:

abcdef!"#$%&

Voit edelleen näppäillä numeroita [CTRL]- ja [CAPS LOCK]-näppäinten ollessa toiminnassa käyttämällä varsinaisen näppäimistön oikealla puolella olevia numeronäppäimiä.

Paluu aikaisempaan toimintatilaan - joko suurten kirjainten tai pienten kirjainten kirjoitustilaan - tapahtuu pitämällä [CTRL]-näppäin alhaalla ja painamalla [CAPS LOCK]-näppäintä kerran. Jos tila, johon palasit, on isojen kirjainten kirjoitustila (CAPS LOCK), paina vielä kerran [CAPS LOCK]-näppäintä, jolloin voit jälleen kirjoittaa pieniä kirjaimia.

[CLR]

Tätä näppäintä käytetään pyyhkimään kohdistimen (eli "kursorin") kohdalla oleva merkki.

Näppäile ABCDEFGH. Kohdistin on nyt viimeisen näppäilemäsi kirjaimen (H) oikealla puolella. Paina nyt kohdistimen vasemmallesiirtonäppäintä [←] neljä kertaa. Kohdistin on siirtynyt neljä merkkipaikkaa vasemmalle ja sen pitäisi olla kirjaimen E päällä.

Huomaat, miten kirjain E yhä näkyy kohdistimen kohdalla. Paina nyt [CLR]-näppäintä kerran ja näet, että kirjain E häviää. Samalla kirjaimet FGH siirtyvät yhden merkkipaikan vasemmalle kirjaimen F näkyessä kohdistimen kohdalla. Paina [CLR]-näppäintä uudelleen ja pidä se alhaalla. Huomaat, kuinka ruudusta häviää ensin kirjain F ja sen jälkeen kirjaimet G ja H.

[ESC]

Tätä näppäintä käytetään haluttaessa poistua siitä toiminnosta, jota tietokone on parhaillaan suorittamassa. Tietokone keskeyttää hetkellisesti toimintansa, kun [ESC]-näppäintä painetaan kerran. Toiminta jatkuu painettaessa jotain muuta näppäintä.

Tietokone poistuu kokonaan parhaillaan suorittamastaan toiminnosta, jos [ESC]-näppäintä painetaan kahdesti. Tällöin tietokone on valmis vastaanottamaan uusia käskyjä näppäimistöltä.

Paina nyt [ESC]-näppäintä kaksi kertaa.

TÄRKEÄÄ!

Kun tulet näyttöruudun oikean reunan kohdalle näppäilemällä ruutuun 40 merkkiä, ilmestyy seuraava merkki automaattisesti seuraavalle riville ruudun vasempaan reunaan. Tämä merkitsee sitä, ETTEI SINUN TARVITSE PAINAA [ENTER]-NÄPPÄINTÄ, kuten ehkä olet tottunut tekemään kirjoittaessasi kirjoituskoneella, jossa on tarpeen suorittaa vaunun palautus lähestyttäessä paperin oikeaa reunaa.

Tietokone tekee tämän automaattisesti puolestasi ja tulostaa ruutuun virheilmoituksen **Syntax error** , jos olet painanut [ENTER]-näppäintä väärässä kohdassa. Virheilmoitus ilmestyy näyttöön joko sillä hetkellä kun virhe tapahtuu tai kun ohjelma ajetaan.

Muotovirheet (Syntax Error)

Jos kuvaruutuun ilmestyy viesti **Syntax error** tarkoittaa se sitä, ettei tietokone ole ymmärtänyt sille antamaasi käskyä. Näppäile esimerkiksi seuraava rivi:

    printt [ENTER]

Tällöin kuvaruutuun ilmestyy viesti:

    Syntax error

Virheilmoitus tulee näyttöön siksi, ettei tietokone ymmärrä antamaasi käskyä **printt** .

Jos näppäilet virheen johonkin ohjelmariviin kuten

    10 printt "abc" [ENTER]

ei virheilmoitus **Syntax error** ilmesty kuvaruutuun ennen kuin tietokone käsittelee käskyn ohjelmaa ajettaessa.

Näppäile:

run [ENTER]

Tällöin ruudussa näkyy:

```
Syntax error in 10
10 printt "abc"
```

Tämä ilmoitus kertoo sinulle sen rivin, jolla virhe on esiintynyt. Samalla näkyy virheellinen ohjelmarivi ja korjailukohdistin, jotta voit korjata virheen.

Paina kohdistimen oikeallesiirtonäppäintä [→], kunnes kohdistin on sanan **printt** viimeisen t-kirjaimen kohdalla. Poista nyt ylimääräinen t painamalla [CLR]-näppäintä, ja syötä sitten korjattu rivi tietokoneen muistiin painamalla [ENTER]-näppäintä.

Näppäile nyt **run [ENTER]** ja huomaat, että tietokone on hyväksynyt antamasi käskyn ja tulostaa ruutuun **abc** .

JOHDATUS AMSTRADIN BASIC-KIELEN AVAINSANOIHIN

Käsittelemme tässä luvussa joitakin yleisimmin käytettyjä AMSTRADin BASIC-kielen avainsanoja. Tarkempi kuvaus kaikista BASIC-kielen avainsanoista löytyy luvusta 8.

CLS

Näppäile cls (clear screen = tyhjennä ruutu). Voit käyttää joko isoja kirjaimia tai pieniä kirjaimia. Päätä näppäily painamalla [ENTER]-näppäintä. Huomaat ruudun tyhjenevän ja sanan **Ready** ilmestyvän kuvaruudun vasempaan yläreunaan yhdessä kohdistimen (värillisen neliön) kanssa.

PRINT

Tätä käskyä käytetään aina silloin, kun haluat tulostaa ruutuun merkkejä, sanoja tai kuvia. Näppäile seuraava käskyrivi:

```
print "hei" [ENTER]
```

Ruudussa näkyy:

```
hei
```

Lainausmerkkejä "" käytetään ilmoittamaan tietokoneelle, mitä kuvaruutuun pitää tulostaa. Ruutuun ilmestyi sana **hei** heti kun [ENTER]-näppäintä painettiin. Tyhjennä ruutu näppäilemällä **cls [ENTER]** .

RUN

Edellä oli esimerkki ohjelmasta, jossa oli vain yksi rivi. Useimmissa ohjelmissa on kuitenkin monta ohjelmariviä. Jokaisen rivin alkuun näppäillään ensin rivinumero. Rivinumerot kertovat tietokoneelle järjestyksen, jossa ohjelma on suoritettava. [ENTER]-näppäintä painettaessa rivi tallentuu tietokoneen muistiin ja on siellä valmiina ohjelman ajoa varten. Näppäile seuraava rivi:

```
10 print "hei" [ENTER]
```

Huomasit varmaan, ettei sana **hei** tulostunut ruutuun painaessasi [ENTER]-näppäintä. Jotta sana tulostuisi ruutuun, on sinun syötettävä käsky **run** . Näppäile siis seuraava rivi:

```
run [ENTER]
```

Nyt sana **hei** tulostui ruutuun. Voit käyttää myös kysymysmerkkiä **?** sen sijaan, että näppäilisit jatkuvasti sanan **print** . Esimerkki:

```
10 ? "hei" [ENTER]
```

LIST

Kun ohjelma on tallennettu muistiin, voit tarkistaa kaiken sen minkä olet näppäillyt pyytämällä ohjelman listauksen. Tämä tapahtuu näppäilemällä:

```
list [ENTER]
```

Tällöin ruutuun ilmestyy:

```
10 PRINT "hei"
```

Se on juuri muistiin tallennettu ohjelma.

Pane merkille, että sana **PRINT** on nyt tulostunut isoilla kirjaimilla. Tämä merkitsee sitä, että tietokone on hyväksynyt sanan **PRINT** tunnettuna BASIC-avainsanana.

Tyhjennä ruutu näppäilemällä **cls [ENTER]** . Huomaa, että vaikka ruutu tyhjeneekin näppäillessäsi **cls [ENTER]** , ei ohjelma pyyhkiydy pois tietokoneen muistista.

GOTO

Tämä avainsana ilmoittaa tietokoneelle, että sen on mentävä joltakin riviltä jollekin toiselle riville joko siksi, että ohjelmassa halutaan ohittaa tietyt ohjelmarivit tai ohjelmaan halutaan muodostaa silmukka. Näppäile:

```
10 print "hei" [ENTER]
20 goto 10 [ENTER]
```

Näppäile:

```
run [ENTER]
```

Näet nyt kuinka sana **hei** tulostuu jatkuvasti kuvaruutuun allekkain ruudun vasempaan reunaan. Voit pysäyttää ohjelman painamalla [ESC]-näppäintä yhden kerran. Ohjelma käynnistyy uudelleen, kun painat jotain muuta näppäintä. Jos haluat pysäyttää ohjelman voidaksesi näppäillä koneeseen muita käskyjä, paina [ESC]-näppäintä kahdesti.

Tyhjennä ruutu näppäilemällä

```
cls [ENTER]
```

Seuraavaksi haluamme, että sana **hei** tulostuu jatkuvasti joka riville sana toisensa perään siten, että koko kuvaruutu tulee täyteen. Tämä tapahtuu seuraavan ohjelman avulla, jossa käytetään puolipistettä **;** lainausmerkin **"** jälkeen. Näppäile:

```
10 print "hei"; [ENTER]
20 goto 10 [ENTER]
run [ENTER]
```

Puolipiste ; ilmoittaa siis tietokoneelle, että sen on tulostettava seuraava merkki välittömästi edellisen merkin perään. Voit poistua tästä ohjelmasta painamalla [ESC]-näppäintä kahdesti. Kirjoita seuraavaksi rivi 10 uudelleen, mutta käytä nyt tavallista pilkkua , puolipisteen ; tilalla.

```
10 print "hei", [ENTER]
run [ENTER]
```

Kuten kuvasta huomaat, on pilkku , ilmoitus tietokoneelle, että haluat tietokoneen tulostavan seuraavan merkin (tai merkkiryhmän) kolmentoista sarakkeen päähän ensimmäisen merkkiryhmän alkukohdasta. Voit käyttää tätä ominaisuutta hyväksesi tietojen esittämiseen erillisissä sarakkeissa. Huomaa kuitenkin, että jos merkkiryhmässä olevien merkkien lukumäärä on suurempi kuin 12, siirtyy seuraava merkki eteenpäin toiset kolmetoista saraketta.

Poistu jälleen ohjelmasta painamalla [ESC]-näppäintä kahdesti. Nollaa tietokoneen muisti aivan kokonaan pitämällä [SHIFT]-, [CTRL]- ja [ESC]-näppäimiä alhaalla tässä järjestyksessä. Tällöin tietokone palaa takaisin alkutilaan.

INPUT

Tätä käskyä käytetään ilmoittamaan tietokoneelle, että tietokoneeseen aiotaan syöttää jotain tietoja sisään - esimerkiksi vastaus esitettyyn kysymykseen. Näppäile seuraava ohjelma:

```
10 input "Kuinka vanha olet"; vuodet [ENTER]
20 print "Olet nuorekkaampi kuin";vuodet
 "vuotias." [ENTER]
run [ENTER]
```

Ruudussa näkyy:

```
Kuinka vanha olet?
```

Näppäile oma ikäsi ja paina lopuksi [ENTER]-näppäintä. Jos ikäsi on esim. 18 vuotta, ilmestyy ruutuun seuraava teksti:

```
Olet nuorekkaampi kuin 18 vuotias.
```

Edellä oleva esimerkki näyttää, miten INPUT-käskyä ja numeromuuttujaa käytetään. Sana **vuodet** laitettiin muistiin rivin 10 lopussa, jotta tietokone yhdistäisi sanan **vuodet** mihin hyvänsä lukuun joka tietokoneeseen syötetään ja tulostaisi tämän luvun samaan kohtaan, jossa sana **vuodet** esiintyy rivillä 20. Vaikka käytimmekin edellä olevassa esimerkissä sanaa **vuodet** ikää kuvaavana muuttujana, olisimme aivan yhtä hyvin voineet käyttää jotakin kirjainta esim. kirjainta **b** .

Nollaa tietokoneen muisti painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä. Jos tietokoneeseen halutaan syöttää mitä hyvänsä merkkejä (kirjaimia tai kirjaimia ja numeroita) input-lauseessa, täytyy muuttujan viimeisenä merkkinä käyttää dollarin merkkiä **$** .

Näppäile koneeseen seuraava ohjelma:
(Huomaa, että sinun on laitettava yksi tyhjä väli sanan **Hei** jälkeen ja ennen **minun** sanaa).

```
10 input "Kuka olet (nimi)"; nimi$ [ENTER]
20 print "Hei "; nimi$" minun nimeni on
Arnold." [ENTER]
run [ENTER]
```

Ruudussa näkyy:

```
Kuka olet (nimi)?
```

Näppäile nimesi ja paina sitten [ENTER]-näppäintä.
Jos nimesi on vaikkapa Pertti, näkyy ruudussa seuraava teksti:

```
Hei Pertti minun nimeni on Arnold.
```

(Lienee syytä mainita, että **ARNOLD** oli AMSTRAD CPC 464:n koodinimi sen kehittelyvaiheen aikana).

Vaikka käytimmekin edellä olevassa esimerkissä nimen merkkijonomuuttujana ilmausta **nimi$** , olisimme aivan yhtä hyvin voineet käyttää jotakin kirjainta esimerkiksi **a$** . Yhdistäkäämme nyt edellä olevat kaksi esimerkkiä yhdeksi ohjelmaksi.

Nollaa tietokoneen muisti painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä. Näppäile koneeseen seuraava ohjelma:

```
5 cls [ENTER]
10 input "Kuka olet (nimi)"; a$ [ENTER]
20 input "Kuinka vanha olet"; b [ENTER]
30 print "Minun on sanottava sinulle ";a
$ " ettet ole";b"vuotiaan nakoinen." [ENTER]
run [ENTER]
```

Olemme käyttäneet tässä ohjelmassa kahta muuttujaa: **a$** kuvaa nimeä ja **b** kuvaa ikää. Käynnistettyäsi ohjelman ruudussa näkyy:

```
Kuka olet (nimi)?
```

Näppäile nyt oma nimesi (esim. Pertti) ja paina sitten [ENTER]-näppäintä. Tämän jälkeen kone kysyy sinulta:

```
Kuinka vanha olet?
```

Näppäile nyt oma ikäsi (esim. 18) ja paina sitten [ENTER]-näppäintä. Jos nimesi olisi Pertti ja olisit 18-vuotias, näkyisi ruudussa seuraava teksti:

```
Minun on sanottava sinulle Pertti
 ettet ole 18 vuotiaan nakoinen.
```

OHJELMAN KORJAILU ELI EDITOINTI

Jos olet näppäillyt väärin jonkun ohjelmarivin ja siitä on seurauksena muotovirheilmoitus (Syntax error) tai jokin muu virheilmoitus, voit korjata kyseisen ohjelmarivin sen sijaan että näppäilisit sen kokonaan uudelleen. Harjoittelemme tätä näppäilemällä edellisen ohjelman virheellisessä muodossa.

Näppäile koneeseen seuraava ohjelma:

```
5 clss [ENTER]
10 input "Kuka olet (nimi)"; a$ [ENTER]
20 input "Kuinka vaha olet"; b [ENTER]
30 print "Minun on sanottava sinulle"; a$
 " ettet ole";b"vuotiaan nakoinen." [ENTER]
```

Yllä olevassa ohjelmassa on kolme virhettä:

Riville 5 on näppäilty **clss** , vaikka siinä pitäisi olla **cls** .
Riville 20 on näppäilty **vaha** , vaikka siinä pitäisi olla **vanha** .
Riviltä 30 puuttuu yksi välilyönti sanan **sinulle** ja lainausmerkin **"** välistä. (Huomaa, että se ei ole virhe, ettei kuvaruudussa näy ä- ja ö-kirjaimia, koska ne eivät kuulu CPC 464:n vakiomerkkivalikoimaan. Opit myöhemmin määrittelemään grafiikkamerkkejä, ja voit käyttää sitä taitoa vaikkapa juuri näiden puuttuvien kirjainten tulostukseen).

Ohjelman korjaamiseen on käytettävissä kolme tapaa. Ensimmäinen näistä on se, että uusi rivi yksinkertaisesti näppäillään uudelleen. Kun rivi näppäillään uudelleen ja syötetään muistiin painamalla [ENTER]-näppäintä, korvaa se muistissa olevan samalla rivinumerolla varustetun rivin. Toinen tapa korjailla ohjelmaa on käyttää korjailukäskyä **edit** , ja kolmas tapa on kopiointikohdistimen (copy cursor) käyttö.

Korjailukäskyn (edit) käyttö

Rivillä 5 oleva virhe korjataan näppäilemällä:

```
edit 5 [ENTER]
```

Nyt rivi 5 tulostuu ruutuun alimmaiseksi ja kohdistin on numeron 5 päällä. Ylimääräinen **s** poistetaan sanasta **clss** painamalla kohdistimen oikeallesiirtonäppäintä [→] kunnes kohdistin on viimeisen s-kirjaimen kohdalla, jossa painetaan [CLR]-näppäintä. Huomaat, että s-kirjain katoaa. Paina nyt [ENTER]-näppäintä, jolloin rivi 5 tallentuu muistiin oikeassa muodossa. Voit tarkistaa tämän näppäilemällä:

```
list [ENTER]
```

Kopiointikohdistimen (copy cursor) käyttö

Rivillä 20 oleva virhe korjataan pitämällä [SHIFT]-näppäin alhaalla ja painamalla sitten kohdistimen ylöspäinsiirtonäppäintä [↑] kunnes kopiointikohdistin on aivan rivin 20 alussa. Huomaat, ettei varsinainen kohdistin ole liikkunut mihinkään, joten kuvaruudussa näkyy nyt kaksi kohdistinta. Paina nyt [COPY]-näppäintä, kunnes kopiointikohdistin on sanan **vaha** h-kirjaimen kohdalla. Huomaat, kuinka rivi 20 kirjoittuu uudelleen kuvaruudun viimeiseksi riviksi ja varsinainen kohdistin pysähtyy samaan kohtaan kuin kopiointikohdistinkin. Näppäile nyt kirjain **n** ja havaitset, että se ilmestyy vain alimmalle riville. Varsinainen kohdistin on siirtynyt, mutta kopiointikohdistin ei ole liikkunut mihinkään. Paina seuraavaksi [COPY]-näppäintä, kunnes koko rivi 20 on kirjoittunut ruudun alareunaan. Kun nyt painat [ENTER]-näppäintä, tallentuu rivi 20 korjatussa muodossa muistiin. Kopiointikohdistin häviää ruudusta ja varsinainen kohdistin siirtyy uuden rivin 20 alle.

Rivillä 30 oleva toinen virhe korjataan pitämällä [SHIFT]-näppäin alhaalla ja painamalla kohdistimen ylöspäinsiirtonäppäintä [↑] kunnes kopiointikohdistin on aivan rivin 30 alkukohdassa. Paina nyt [COPY]-näppäintä, kunnes kopiointikohdistin on sanan **sinulle** perässä olevan lainausmerkin kohdalla. Paina nyt välilyöntinäppäintä yhden kerran. Huomaat, kuinka alimmalle riville ilmestyy yksi välilyönti lisää. Paina seuraavaksi taas [COPY]-näppäintä ja pidä sitä alhaalla, kunnes rivin 30 viimeinenkin merkki on kirjoittunut kuvaruudun alareunaan. Voit nyt painaa [ENTER]-näppäintä, jolloin korjattu rivi tallentuu muistiin. Lopuksi voit tarkistaa, että ohjelma on tallentunut muistiin oikeassa muodossa näppäilemällä:

list [ENTER]

Nollaa lopuksi tietokoneen muisti painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä.

IF THEN

Laajennamme nyt aikaisempaa ohjelmaa käyttämällä IF- ja THEN-käskyjä.

Näppäile koneeseen seuraava ohjelma. Huomaa, että ohjelmaan on lisätty kaksi uutta merkkiä. Merkki < tarkoittaa pienempi kuin, ja se löytyy M-näppäimen oikealta puolelta. Merkki > tarkoittaa suurempi kuin, ja se löytyy edellisen näppäimen oikealta puolelta.

```
5 cls [ENTER]
10 input "Kuka olet (nimi)";a$ [ENTER]
20 input "Kuinka vanha olet";vuodet [ENTER]
30 if vuodet < 13 then 60 [ENTER]
40 if vuodet < 20 then 70 [ENTER]
50 if vuodet > 19 then 80 [ENTER]
60 print "Et siis "a$" ole viela teini-i
kainen, koska olet"vuodet"vuotias.":end [ENTER]
70 print "Ahaa "a$" olet siis teini-ikai
nen, koska olet"vuodet"vuotias.":end [ENTER]
80 print "Et siis "a$" ole enaa teini-ik
ainen, koska olet"vuodet"vuotias." [ENTER]
```

Tarkista ohjelma näppäilemällä:

```
list [ENTER]
```

Näppäile seuraavaksi:

```
run [ENTER]
```

Vastaa nyt tietokoneen esittämiin kysymyksiin ja katso mitä tapahtuu.

Havaitset, mikä vaikutus IF- ja THEN-käskyillä on ohjelmassa. Rivien 60 ja 70 loppuun on myös lisätty sana **end** . Tätä avainsanaa käytetään lopettamaan ohjelman kulku. Jos sanaa **end** ei olisi rivin 60 lopussa, jatkaisi tietokone ohjelman suorittamista rivin 60 jälkeenkin ja suorittaisi rivit 70 ja 80.

Aivan samalla tavalla sanan **end** puuttuminen rivin 70 lopussa aiheuttaisi sen, että tietokone jatkaisi ohjelman suorittamista rivin 70 jälkeenkin ja suorittaisi saman tien rivin 80. Kaksoispiste : end-sanan edessä erottaa sen edellisestä käskystä. Voit käyttää kaksoispisteitä : halutessasi laittaa kaksi tai useampia käskyjä samalle ohjelmariville. Ohjelman alkuun on myös lisätty rivi 5, jolla ruutu tyhjennetään ohjelman käynnistyessä. Käytämme tästä eteenpäin samanlaista ohjelmariviä ohjelmien alussa saadaksemme ohjelmat käyttäytymään hieman siistimmin.

Muita IF- ja THEN-käskyihin liittyviä avainsanoja ovat ELSE, OR ja GOTO. Näiden sanojen käyttö IF-käskyn kanssa selviää myöhemmin, kunhan maltat käydä käsikirjaa läpi järjestelmällisesti.

Nollaa nyt tietokoneen muisti painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä.

FOR TO NEXT

Käsitelkäämme nyt FOR-, TO- ja NEXT-käskyjen käyttöä. Seuraavassa esimerkissä on näytetty, kuinka tietokone tulostaa luvun 12 kertotaulun (1x12, 2x12, 3x12, jne.). Näppäile koneeseen seuraava ohjelma. Merkki * tarkoittaa kertolaskua.

```
5 cls [ENTER]
10 for a = 1 to 20 [ENTER]
20 print a"* 12 ="a*12 [ENTER]
30 next a [ENTER]
run [ENTER]
```

Sarakkeet eivät ole oikein siistin näköiset, joten näppäile seuraava ohjelma:

```
5 cls [ENTER]
10 for a = 1 to 9 [ENTER]
20 print a" * 12 ="a*12 [ENTER]
30 next a [ENTER]
40 for a = 10 to 20 [ENTER]
50 print a"* 12 ="a*12 [ENTER]
60 next a [ENTER]
run [ENTER]
```

Kokeile tätä kertotauluohjelmaa muillakin luvuilla. Jos haluat nähdä esimerkiksi luvun 17 kertotaulun, vaihda riveillä 20 ja 50 olevien lukujen 12 paikalle luku 17. Nollaa lopuksi tietokoneen muisti painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä.

FOR-, TO- ja NEXT-käskyillä toteutettavan silmukan askelväli on mahdollista määritellä STEP-käskyllä. Tarkemmat tiedot löytyvät luvusta 8, jossa käsitellään FOR-käskyä.

YKSINKERTAISTA ARITMETIIKKAA

Voit käyttää CPC 464-tietokonettasi helposti myös laskimena.

Tämä selvinnee parhaiten muutaman esimerkin avulla. Esimerkeissä on käytetty kysymysmerkkiä **?** sen sijaan, että joka kerta erikseen näppäiltäisiin sana **print** . Tulos ilmestyy kuvaruutuun heti kun [ENTER]-näppäintä painetaan.

Yhteenlasku

(Plusmerkki löytyy painamalla näppäimiä [SHIFT] ja ;).

Näppäile:

```
?3+3 [ENTER]
 6
```

(Sinun ei tarvitse näppäillä yhtäläisyysmerkkiä =).

Näppäile:

```
?8+4 [ENTER]
 12
```

Vähennyslasku

(Miinusmerkki löytyy samasta näppäimestä kuin yhtäläisyysmerkkikin =).

Näppäile:

```
?4-3 [ENTER]
 1
```

Näppäile:

```
?8-4 [ENTER]
 4
```

Kertolasku

(Kertomerkki * löytyy painamalla näppäimiä [SHIFT] ja :).

Näppäile:

```
?3*3 [ENTER]
 9
```

Näppäile:

```
?8*4 [ENTER]
 32
```

Jakolasku

(Jakolaskun merkki / löytyy samasta näppäimestä kuin kysymysmerkkikin ?).

Näppäile:

```
?3/3 [ENTER]
 1
```

Näppäile:

```
?8/4
 2
```

Neliöjuuri

Voit laskea jonkun luvun neliöjuuren näppäilemällä **sqr( )** . Luku, josta neliöjuuri halutaan, näppäillään sulkeiden sisään.

Näppäile:

```
?sqr(16) [ENTER]   (tämä tarkoittaa samaa kuin √16)
 4
```

Näppäile:

```
?sqr(100) [ENTER]
 10
```

Potenssiin korottaminen

(Potenssiinkorotusmerkki ↑ löytyy samasta näppäimestä kuin £ -merkkikin).

Potenssiin korottaminen tarkoittaa sitä, että joku luku korotetaan jonkun toisen luvun osoittamaan potenssiin. Esimerkiksi 3 toiseen potenssiin (3^2), 3 kolmanteen potenssiin (3^3), jne. Näppäile:

```
?3↑3 [ENTER]   (tämä tarkoittaa 3³)
 27
```

Näppäile:

```
?8↑4 [ENTER]   (tämä tarkoittaa 8⁴)
 4096
```

Kuutiojuuri

Kuutiojuurien laskeminen käy aivan yhtä helposti kuin neliöjuurien laskeminenkin.

Jos haluat tietää esim. luvun 27 kuutiojuuren ($\sqrt[3]{27}$), näppäile:

```
?27↑(1/3) [ENTER]
 3
```

Luvun 125 kuutiojuuri löytyy näppäilemällä:

```
?125↑(1/3) [ENTER]
 5
```

Yhteen-, vähennys-, kerto- ja jakolaskujen yhdistelmät

Tietokone ymmärtää myös peruslaskutoimitusten yhdistelmätehtäviä, mutta se laskee ne tietyssä tärkeysjärjestyksessä.

Ensimmäisenä tärkeysjärjestyksessä on kertolasku, sitten jakolasku, seuraavaksi yhteenlasku ja viimeisenä on vähennyslasku. Tämä tärkeysjärjestys pätee vain sellaisiin laskutoimituksiin, joissa on pelkästään edellä mainittuja neljää peruslaskutoimitusta. Tärkeysjärjestystä käsitellään myöhemmin lisää, ja siihen lisätään potenssiin korottaminen yms.

Jos laskutehtävämme olisi:

```
3+7-2*7/4
```

Saattaisi luulla, että tietokone laskee tämän seuraavasti:

```
3+7-2 *7 /4
=  8 *7 /4
=    56/4
=    14
```

Tietokone laskee sen kuitenkin seuraavasti:

```
3+7-2*7/4
=3+7-14/4
=3+7-3.5
=10-3.5
=6.5
```

Voit kokeilla tämän näppäilemällä laskutoimituksen:

```
?3+7-2*7/4 [ENTER]
 6.5
```

Voit muuttaa tietokoneen laskujärjestystä lisäämällä sulkumerkkejä. Tietokone käsittelee sulkumerkkien sisällä olevat laskutoimitukset ennen sulkumerkkien ulkopuolella olevia kertolaskuja tai muita laskutoimituksia. Kokeile tätä näppäilemällä sulkumerkit sisältävä laskutoimitus:

```
?(3+7-2)*7/4 [ENTER]
 14
```

Potenssiin korotus laajemmassa mielessä

Jos tarvitset laskutoimituksessa hyvin suuria tai hyvin pieniä lukuja, on joskus kätevää käyttää tieteellistä merkintätapaa. Kirjainta **E** käytetään eksponentin merkitsemiseen sellaisille luvuille, joiden kantaluku on 10. Voit käyttää joko pientä kirjainta **e** tai suurta kirjainta **E** .

Esimerkiksi 300 on sama kuin $3x10^2$. Tieteellisen merkintätavan mukaan tämä on 3E2. Vastaavasti 0,03 on sama kuin $3x10^{-2}$, ja tieteellisen merkintätavan mukaan se on 3E-2. Kokeile seuraavia esimerkkejä:

Esimerkki 1. 30 x 10

Näppäile:

```
?30*10 [ENTER]
 300
```

Voit näppäillä saman laskutehtävän myös seuraavasti:

```
?3el*lel [ENTER]
 300
```

Esimerkki 2. 3000 x 1000

Näppäile:

```
?3e3*le3 [ENTER]
 300000
```

Esimerkki 3. 3000 x 0,001

Näppäile:

```
?3e3*le-3 [ENTER]
 3
```

GRAFIIKKA, ÄÄNET JA NÄYTTÖMUODOT

AMSTRAD CPC 464-tietokoneessa on kolme näyttömuotoa (mode), jotka se tuntee nimillä Mode 0, Mode 1 ja Mode 2.

Näyttömuotona on automaattisesti Mode 1, kun virta kytketään tietokoneeseen.

Jotta ymmärtäisit eri näyttömuotojen toiminnan, kytke virta tietokoneeseen ja paina numeronäppäintä **1** . Pidä näppäin alhaalla kunnes kuvaruudussa on kaksi riviä täynnä ykkösiä. Jos nyt lasket yhdellä rivillä olevien ykkösten määrän, huomaat niitä olevan yhdellä rivillä 40. Tämä tarkoittaa sitä että näyttömuodossa 1 (Mode 1) on 40 saraketta. Paina nyt [ENTER]-näppäintä, jolloin kuvaruutuun ilmestyy virheilmoitus **Syntax Error** . Tästä ei kuitenkaan kannata välittää, koska tämä on vain nopea tapa päästä takaisin valmiustilaan, jonka merkiksi ruutuun ilmestyy **Ready** -ilmoitus. Se on merkki siitä, että tietokone on valmis vastaanottamaan seuraavan käskyn.

Näppäile nyt: mode 0 [ENTER]

Huomaat, kuinka merkit ovat nyt suurempia kuvaruudussa. Paina jälleen numeronäppäintä **1** ja pidä se alhaalla, kunnes ruudussa on taas kaksi riviä täynnä ykkösiä. Jos nyt lasket yhdellä rivillä olevien ykkösten lukumäärän, huomaat niitä olevan yhdellä rivillä 20. Tämä tarkoittaa sitä, että näyttömuodossa 0 (Mode 0) on 20 saraketta. Paina taas [ENTER]-näppäintä.

Näppäile nyt: mode 2 [ENTER]

Ja kuten kuvaruudusta näet, ovat merkit kaikkein pienimpiä tässä näyttömuodossa. Jos taas näppäilet kuvaruutuun kaksi riviä ykkösiä, tulee niitä yhdelle riville 80. Toisin sanoen näyttömuodossa 2 (Mode 2) on 80 saraketta. Siis vielä kerran:

Näyttömuoto 0 (Mode 0) = 20 saraketta.
Näyttömuoto 1 (Mode 1) = 40 saraketta.
Näyttömuoto 2 (Mode 2) = 80 saraketta.

Paina lopuksi vielä kerran [ENTER]-näppäintä.

VÄRIT

AMSTRAD CPC 464-tietokoneen värivalikoimaan kuuluu 27 väriä. Nämä näkyvät vihreä/musta-monitorissa (GT 64) erilaisina vihreän sävyinä. Jos laitteistoosi kuuluu monitori GT 64, voit hankkia AMSTRAD MP1 modulaattori/virtalähdeyksikön, jolloin voit käyttää hyväksesi tietokoneen kaikki värimahdollisuudet liittämällä tietokoneen tavalliseen väritelevisioon.

Kuvaruudussa saadaan näkyviin 27:stä käytettävissä olevasta väristä millä hetkellä hyvänsä enintään 16 näyttömuodossa 0 (Mode 0).

Kuvaruudussa saadaan näkyviin 27:stä käytettävissä olevasta väristä millä hetkellä hyvänsä enintään 4 näyttömuodossa 1 (Mode 1).

Kuvaruudussa saadaan näkyviin 27:stä käytettävissä olevasta väristä millä hetkellä hyvänsä enintään 2 näyttömuodossa 2 (Mode 2).

Voit vaihtaa kuvaruudun reuna-alueen väriä BORDER-käskyllä, paperin (eli taustan) väriä PAPER-käskyllä ja kynän väriä PEN-käskyllä toisistaan riippumatta. CPC 464:n kaikki 27 väriä on lueteltu taulukossa 1. Siinä on myös vastaavat INK-käskyn (musteen) viitenumerot.

VÄRIVALIKOIMA

Musteen numero (INK)	Väri/muste	Musteen numero (INK)	Väri/muste
0	Musta	14	Pastellinsininen
1	Sininen	15	Oranssi
2	Kirkkaansininen	16	Vaaleanpunainen
3	Punainen	17	Pastellin-sinipunainen
4	Sinipunainen		
5	Vaaleansininen	18	Kirkkaanvihreä
6	Kirkkaanpunainen	19	Merenvihreä
7	Purppura	20	Kirkkaansinivihreä
8	Kirkkaan-sinipunainen	21	Vaaleanvihreä
		22	Pastellinvihreä
9	Vihreä	23	Pastellin-sinivihreä
10	Sinivihreä		
11	Taivaansininen	24	Kirkkaankeltainen
12	Kellanvihreä	25	Pastellinkeltainen
13	Siniharmaa	26	Kirkkaanvalkoinen

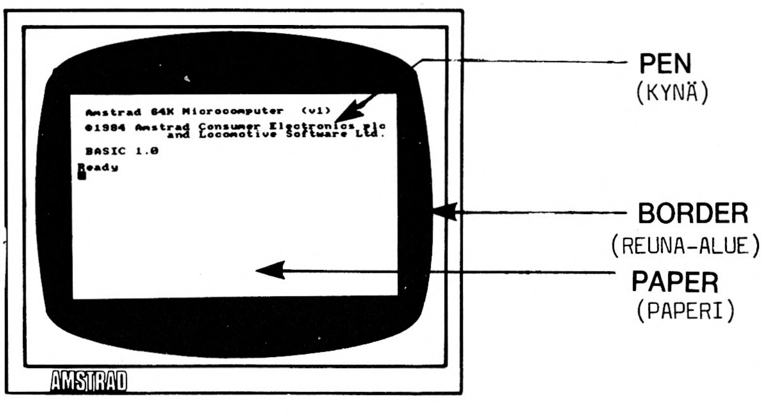
Taulukko 1. Musteen (INK) numerot ja värit

Kuten jo aikaisemmin mainittiin, kytkeytyy tietokone näyttömuotoon 1 (Mode 1), kun virta kytketään tietokoneeseen. Näyttömuotoon 1 palataan takaisin jostakin toisesta näyttömuodosta näppäilemällä:

```
mode 1 [ENTER]
```

Kun virta kytketään tietokoneeseen, ovat reuna-alueen (BORDER), paperin eli taustan (PAPER) ja kynän (PEN) väri:

Reuna-alue (BORDER): väri numero 1 (sininen).
Paperi eli tausta (PAPER): väri numero 1 (sininen).
Kynä (merkit) (PEN): väri numero 24 (kirkkaankeltainen).



Reuna-alueeksi (BORDER) kutsutaan sitä aluetta, joka ympäröi paperia (PAPER). (Huomaa, että sekä reuna-alueen että paperin väri on sininen, kun virta kytketään tietokoneeseen). Paperi on puolestaan se reuna-alueen sisäpuolella oleva ruudun osa, jossa merkit näkyvät. Kynä (PEN) ilmoittaa merkkien värin.

Saadaksemme asiasta selkeämmän kuvan voimme kuvitella mielessämme, että tietokoneen PEN- ja PAPER-käskyjä vastaavat oikea kynä ja pala paperia (engl. pen = suom. kynä ja engl. paper = suom. paperi). Kuvaruudussa näkyvien merkkien väriä voidaan vaihtaa aivan kuten kynän musteenkin (INK) (engl. ink = suom. muste) väriä voidaan vaihtaa. Myös ruudussa näkyvän "paperin" väriä voidaan vaihtaa samalla tavalla kuin oikean paperin väriä voidaan vaihtaa.

Voit vaihtaa reuna-alueen väriä näppäilemällä:

border 0 [ENTER]

Huomaat, kuinka reuna-alueen väri vaihtuu sinisestä mustaksi. Taulukosta 1 näet, että 0 on musta. Voit vaihtaa reuna-alueen värin miksi hyvänsä taulukossa 1 mainituista väreistä näppäilemällä **border** ja sen jälkeen haluamasi värin numeron.

Pyyhi nyt ruutu puhtaaksi näppäilemällä:

cls [ENTER]

Seuraavaksi vaihdamme paperin väriä näppäilemällä:

paper 2 [ENTER]

Sanan **Ready** taustan väri vaihtui kirkkaansiniseksi. Tyhjennä nyt ruutu näppäilemällä **cls [ENTER]** , jolloin ruutu vaihtuu kokonaan uuden paperin väriseksi.

Seuraavaksi vaihdetaan kynän (PEN) väriä näppäilemällä:

pen 3 [ENTER]

Kuten huomaat, on kynän väri vaihtunut ja sana **Ready** on tulostunut ruutuun punaisena. Näppäile lopuksi:

cls [ENTER]

Asiat on ehkä helpompi ymmärtää, jos seuraat myös taulukkoa 2. Kun virta kytketään tietokoneeseen, on paperin numero 0. Taulukon 2 ensimmäisestä sarakkeesta näet kynän numero 0. Jos katsot samalta riviltä näyttömuodon 1 saraketta, löydät siitä värin numero 1.

Värikartta (taulukko 1) puolestaan kertoo, että väri numero 1 on sininen. Ja se on juuri paperin väri silloin, kun virta kytketään tietokoneeseen.

Muutimme juuri äsken paperin numeroksi 2. Kun katsot taulukon 2 vasemman sarakkeen paperin numeron 2 kohdalta näet samalta riviltä näyttömuodon 1 sarakkeen kohdalta, että väri on numero 20. Taulukosta 1 näkyy, että väri numero 20 on kirkkaansinivihreä.

Paperin/kynän numero (PAPER/PEN)	Musteen väri (INK) Näyttömuoto 0 (Mode 0)	Näyttömuoto 1 (Mode 1)	Näyttömuoto 2 (Mode 2)
0	1	1	1
1	24	24	24
2	20	20	1
3	6	6	24
4	26	1	1
5	0	24	24
6	2	20	1
7	8	6	24
8	10	1	1
9	12	24	24
10	14	20	1
11	16	6	24
12	18	1	1
13	22	24	24
14	Vilkkuva 1,24	20	1
15	Vilkkuva 16,11	6	24

Taulukko 2. Paperin, kynän, näyttömuodon ja musteen numeroiden välinen riippuvuussuhde



Kun virta kytketään aluksi tietokoneeseen, käyttää se kynää numero 1. Taulukosta 2 näyttömuodon 1 sarakkeen kohdalta näkyy, että kynän numeroa 1 vastaa väri numero 24. Taulukko 1 puolestaan kertoo, että väri numero 24 on kirkkaankeltainen, joka juuri on ruudussa näkyvien merkkien (kynän) väri silloin kun virta kytketään tietokoneeseen.

Vaihdoimme juuri äsken kynän (PEN) numeroksi 3. Taulukosta 2 näkyy, että kynän numero 3 väri näyttömuodossa 1 on väri numero 6. Taulukosta 1 näkyy puolestaan se, että väri numero 6 on kirkkaanpunainen.

Juuri parhaillaan meillä on käytössä paperi numero 2 ja kynä numero 3. Voimme nyt vaihtaa niiden värit jatkaessamme kokeiluja. Vaihdamme niiden värit käyttämällä INK-käskyä. INK-käskyssä on kaksi numeroa, joista ensimmäinen on vaihdettavan kynän (PEN) tai paperin (PAPER) numero ja toinen on sen värin numero, jonka väriseksi kynä tai paperi vaihdetaan. Katso värien numeroita taulukosta 1. Seuraavassa esimerkissä näytetään, kuinka paperin 2 väri vaihdetaan mustaksi ja kynän 3 väri vaihdetaan kirkkaanvalkoiseksi.

Taulukosta 1 näkyy, että mustan värin numero on 0 ja kirkkaanvalkoisen värin numero on 26. Näppäile nyt:

ink 2,0 [ENTER]

(Kuten äsken tuli ilmi, on 2 tämänhetkisen paperin numero ja 0 on musta).

Näppäile nyt seuraava rivi:

ink 3,26 [ENTER]

(3 on tämänhetkisen kynän numero ja 26 on kirkkaanvalkoinen).

Nollaa nyt tietokone aivan kokonaan painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä.

Kuten aikaisemmin tuli jo esille, on paperin numero 0 ja kynän numero 1 silloin kun virta kytketään aluksi tietokoneeseen tai kun tietokone nollataan painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä. Paperin väri on 1 (sininen) ja kynän väri on 24 (kirkkaankeltainen). Nämä voitaisiin myös näppäillä **ink 0,1** paperia varten (eli paperin 0 väriksi asetetaan väri numero 1) ja **ink 1,24** kynää varten (eli kynän 1 väriksi asetetaan väri numero 24). Jos nämä halutaan vaihtaa välittömästi valkoisiksi merkeiksi (kynäksi) mustalla pohjalla (paperilla), näppäillään ensin **ink 0,0 [ENTER]** ja seuraavaksi **ink 1,26 [ENTER]** .

VILKKUVAT VÄRIT

Kuvaruudussa näkyvien merkkien väri saadaan vilkkumaan kahden valitun värin välillä. Tämä saadaan aikaan lisäämällä yksi värinumero lisää kynää (PEN) koskevaan ink-käskyyn.

Seuraavassa esimerkissä haluamme saada ruudussa näkyvät merkit vilkkumaan kirkkaanvalkoisen ja kirkkaanpunaisen välillä. Näppäile siis seuraava rivi:

ink 1,26,6 [ENTER]

Tässä tapauksessa 1 on kynän numero, 26 on kirkkaanvalkoisen värin numero ja 6 on vuorottelevan värin eli kirkkaanpunaisen värin numero.

Myös merkkien taustana olevan paperin väri on mahdollista saada vilkkumaan kahden värin välillä. Tämä saadaan aikaan lisäämällä yksi värinumero lisää paperia (PAPER) koskevaan ink-käskyyn.

Seuraavassa esimerkissä haluamme saada paperin värin vilkkumaan vihreän ja kirkkaankeltaisen välillä ruutuun tulostuvien merkkien takana. Siispä näppäile seuraava rivi:

ink 0,9,24 [ENTER]

Tässä tapauksessa 0 on paperin numero, 9 on vihreän värin numero ja 24 on vuorottelevan värin eli kirkkaankeltaisen värin numero.

Nollaa nyt tietokone painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä.

Huomasit jo ehkä taulukosta 2, että kaksi kynän numeroa (numerot 14 ja 15) yhdessä kahden paperin numeron (numeroiden 14 ja 15) kanssa ovat vilkkuvia värejä näyttömuodossa 0. Toisin sanoen ink-käskyyn ei ole tarpeen lisätä enää ylimääräistä numeroa. Näppäile seuraavat kaksi riviä:

mode 0 [ENTER]
pen 15 [ENTER]

Tällöin ruutuun ilmestyy sana **Ready** , joka vilkkuu taivaansinisen ja vaaleanpunaisen välillä. Näppäile seuraavaksi:

paper 14 [ENTER]
cls [ENTER]

Huomaat, kuinka sana **Ready** vilkkuu edelleen taivaansinisen ja vaaleanpunaisen välillä, mutta sen lisäksi vilkkuu myös paperi (tausta) keltaisen ja sinisen välillä.

Näitä tietokoneen oletusarvoina valitsemia vilkkuvia värejä on mahdollista vaihtaa näppäilemällä uusi kynää (PEN) tai paperia (PAPER) koskeva ink-käsky. Esimerkiksi kynän väri saadaan vaihdetuksi vilkkumaan mustan ja kirkkaanvalkoisen välillä näppäilemällä:

ink 15,0,26 [ENTER]

Edellisellä sivulla olevassa esimerkissä 15 tarkoittaa kynän numeroa, 0 on mustan värin numero ja 26 on vuorottelevan värin eli kirkkaanvalkoisen värin numero.

Ja ei kahta ilman kolmatta: myös reuna-alueen (BORDER) väri on mahdollista saada vilkkumaan kahden värin välillä lisäämällä BORDER-käskyyn yksi ylimääräinen värinumero. Kokeile tätä näppäilemällä:

```
border 6,9 [ENTER]
```

Ja kuten näet, vilkkuu reuna-alueen väri kirkkaanpunaisen ja vihreän välillä.

Nollaa nyt tietokone aivan kokonaan painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä.

ESITTELYOHJELMA

Käytettävissä olevia värejä esitellään lisää seuraavassa esittelyohjelmassa. Näppäile ohjelma koneeseen ja aja se run-käskyllä. Ohjelmaan on lisätty joitakin ääniä, mutta niitä käsitellään hiukan myöhemmin tässä luvussa.

```
10 mode 0: ink 0,2:ink 1,24:paper 0 [ENTER]
20 pen 1: for b=0 to 26: border b [ENTER]
30 locate 3,12:print"BORDER COLOUR";b [ENTER]
40 sound 4,(40-b) [ENTER]
50 for t=1 to 600:next t:next b:cls [ENTER]
60 for p=0 to 15:paper p:pen 5:print "pa
per" ;p:print [ENTER]
70 for n=0 to 15:pen n:print "pen";n [ENTER]
80 sound 1,(n*20+p) [ENTER]
90 for t=1 to 100:next t:next n [ENTER]
100 for t=1 to 1000:next t:cls:next p [ENTER]
110 cls:paper 0:pen 1:locate 7,12:print
"THE END": for t=1 to 2000: next t [ENTER]
120 mode 1: border 1:ink 0,1:ink 1,24:pa
per 0:pen 1 [ENTER]

run [ENTER]
```

Ohjelmassa:

border = reuna-alue
paper = paperi
pen = kynä
colour = väri

(Voit pysäyttää ohjelman missä kohdassa hyvänsä painamalla [ESC]-näppäintä. Ohjelma jatkuu painaessasi jotain muuta näppäintä).

GRAFIIKKA

Tästä eteenpäin emme pyydä sinua painamaan [ENTER]-näppäintä jokaisen ohjelmarivin lopussa. Oletamme, että teet sen aivan automaattisesti. Tietokoneen muistiin on ohjelmoitu koko joukko merkkien symboleja. Saat minkä hyvänsä näistä merkeistä näkyviin käyttämällä avainsanaa **chr$( )**. Sulkumerkkien sisään laitetaan symbolin numero, joka voi olla mikä hyvänsä välillä 32 - 255.

Nollaa tietokone painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä, ja näppäile sitten:

```
print chr$(250)
```

Tällöin ruutuun ilmestyy merkki numero 250, joka on oikealle kävelevä mies.

Jos haluat nähdä ruudussa kaikki tietokoneen muistiin ohjelmoidut merkit ja symbolit niiden viitenumeroineen, näppäile koneeseen seuraava ohjelma ja muista painaa [ENTER]-näppäintä jokaisen rivin lopussa.

```
10 for n = 32 to 255: print n;chr$(n);
20 next n
run
```

Voit vertailla ruutuun tulostuneita merkkejä tämän käsikirjan takaosassa olevaan liitteeseen III. Sieltä löytyvät vastaavat merkit viitenumeroineen.

LOCATE

Tätä käskyä käytetään merkkikohdistimen sijoittamiseksi uudelleen tiettyyn kuvaruudun kohtaan. Merkkikohdistimen alkukohta on aina kuvaruudun vasemmassa yläkulmassa, ellei sitä muuteta locate-käskyllä. Oletusarvona käytettävän alkukohdan x- ja y-koordinaatit ovat 1 ja 1 (x on vaakasuunnan arvo ja y on pystysuunnan arvo). Näyttömuodossa 1 (mode 1) on 40 saraketta ja 25 riviä. Jos haluamme sijoittaa merkin ylimmän rivin keskelle näyttömuodossa 1, on x- ja y-koordinaatteina käytettävä arvoja 20 ja 1.

Kokeilkaamme tätä näppäilemällä seuraava ohjelma (jokaisen rivin lopussa on muistettava painaa [ENTER]-näppäintä).

```
mode 1              ...ruutu tyhjenee ja kohdistin siirtyy vasempaan
                    ylänurkkaan

10 locate 20,1
20 print chr$(250)
run
```

Voit varmistua siitä, että merkki tulostui ylimmälle riville näppäilemällä:

```
border 0
```

Reuna-alue muuttuu nyt mustaksi ja näet ruudun ylimmän rivin keskellä miestä esittävän merkin.

Näyttömuodossa 0 (mode 0) on vain 20 saraketta mutta samat 25 riviä. Näppäile:

```
mode 0
run
```

Huomaat, kuinka mies ilmestyy nyt kuvaruudun oikeaan ylänurkkaan. Näin käy siksi, että x-koordinaatti 20 on viimeinen sarake näyttömuodossa 0.

Näyttömuodossa 2 (mode 2) on 80 saraketta ja edelleen 25 riviä. Arvaat varmaankin mihin mies ilmestyy, jos käytät samaa kokeiluohjelmaa kuin edellä. Näppäile siis:

```
mode 2
run
```

Palaa takaisin näyttömuotoon 1 näppäilemällä **mode 1** .

Suorita nyt omia kokeiluja muuttamalla locate- ja chr$( )-käskyjen numeroarvoja, joilla saat merkin tulostumaan mihin hyvänsä kohtaan näyttöruudussa. Seuraava rivi käyköön esimerkistä:

```
locate 20,12:print chr$(240)
```

Kuvaruudun keskelle ilmestyy nyt nuoli. Pane merkille, että tässä käskyssä:

20 on vaakasuunnan koordinaatti (x) (väliltä 1 - 40)
12 on pystysuunnan koordinaatti (y) (väliltä 1 - 25)
240 on merkin numero (väliltä 32 - 255).

Seuraava ohjelma aiheuttaa sen, että merkki nro 250 tulostuu toistuvasti ruudun yli. Näppäile ohjelma koneeseen ja kokeile sitä.

```
5 cls
10 for x = 1 to 39
20 locate x,20
40 print chr$(250)
50 next x
60 goto 5
run
```

Ohjelma katkeaa, kun painat [ESC]-näppäintä kahdesti.

Jos haluat, että edellinen merkki häviää ruudusta ennen kuin seuraava merkki tulostuu ruutuun, näppäile riviksi 40:

```
40 print " "; chr$(250)
```

(Tämä uusi rivi 40 korvaa automaattisesti aiemmin näppäillyn rivin 40).

Käynnistä ohjelma näppäilemällä **run** .

Merkin liike ruudun yli paranee, jos lisäät ohjelmaan seuraavan rivin:

```
30 call &bd19
```

Ohjelmaa voidaan parantaa entisestään lisäämällä siihen joitakin viivästyssilmukoita ja käyttämällä erilaista paluusuunnan merkkiä. Näppäile ensin **list** , ja lisää sitten ohjelmaan seuraavat rivit:

```
60 for n = 1 to 300:next n
65 for x = 39 to 1 step -1
70 locate x,20
75 call &bd19
80 print chr$(251);" "
85 next x
90 for n = 1 to 300:next n
95 goto 10
run
```

Seuraavaksi olemme laatineet kokeiltavaksesi pienen mielenkiintoisen ohjelman. Siihen on lisätty eräitä uusia käskyjä, joita käsitellään hiukan myöhemmin. Älä välitä niistä, vaan näppäile ohjelma koneeseen ja kokeile sitä.

```
new

10 mode 1
20 locate 21,14:print chr$(244)
30 tag
40 for x=0 to 624 step 2
50 mover -16,0
60 if x<308 or x>340 then y=196:goto 90
70 if x<324 then y=x-104:goto 85
80 y=536-x
85 sound 1,0,20,7
90 move ox, oy:print " ";:ox=x:oy=y
100 move x,y
110 if (x mod 4) = 0 then print chr$(250
); else print chr$(251);
120 for n=1 to 4: call &bd19:next n
130 next x
140 tagoff
150 goto 20
run
```

PLOT

Tätä käskyä käytetään locate-käskystä poiketen määrittelemään grafiikkakohdistimen paikka käyttämällä kuva-alkiokoordinaatteja. (Kuva-alkio (pixel) on erittäin pieni kuvaruudun osa).

Grafiikkakohdistin ei näy ruudussa ja se on erilainen kuin merkkikohdistin.

Kuva-alkioiden määrä on 640 x 400 (640 vaakasuunnassa ja 400 pystysuunnassa). X- ja Y-koordinaattien paikka ilmoitetaan suhteessa kuvaruudun vasempaan alanurkkaan, jonka X- ja Y-koordinaatit ovat 0 ja 0. Grafiikkakohdistimen X- ja Y-koordinaatit eivät eroa näyttömuodoissa 0, 1 ja 2 toisin kuin locate-käskyllä, jota käytetään merkkien kohdistamiseen.

Asia selviää parhaiten kokeilemalla. Nollaa ensin tietokone painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä, ja näppäile sitten alla oleva rivi (ja muista painaa [ENTER]-näppäintä rivin lopussa):

```
plot 320,200
```

Ruudun keskelle ilmestyy pieni piste.
Vaihda nyt näyttömuotoa (mode) näppäilemällä:

```
mode 0
plot 320,200
```

Huomaat, kuinka piste on edelleen ruudun keskellä, mutta piste on nyt suurempi. Vaihda jälleen näyttömuotoa ja näppäile sama käsky, jotta näet sen vaikutuksen näyttömuodossa 2. Näppäile siis seuraavasti:

```
mode 2
plot 320,200
```

Piste on edelleen ruudun keskellä, mutta nyt se on paljon pienempi.

Sijoita useampia pisteitä ruudun eri kohtiin erilaisissa näyttömuodoissa käyttäen plot-käskyä, jotta käsky tulisi sinulle tutuksi. Palaa lopuksi takaisin näyttömuotoon 1 ja tyhjennä ruutu näppäilemällä **mode 1** .

DRAW

Nollaa ensin tietokone painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä. Käskyllä **draw** voidaan piirtää suora viiva tämänhetkisestä grafiikkakohdistimen paikasta. Asia selviää tarkemmin, kun piirrämme kuvaruutuun suorakaiteen seuraavalla ohjelmalla. Aluksi sijoitamme grafiikkakohdistimen uuteen kohtaan plot-käskyllä. Seuraavaksi piirretään viiva grafiikkakohdistimen paikasta ylöspäin vasenta yläkulmaa kohden, siitä jatketaan oikeaan kulmaan, jne.

Näppäile:

```
5 cls
10 plot 10,10
20 draw 10,390
30 draw 630,390
40 draw 630,10
50 draw 10,10
60 goto 10
run
```

Katkaise ohjelma painamalla [ESC]-näppäintä kahdesti.

Lisää nyt seuraavat rivit edellä tekemääsi ohjelmaan, jolloin ohjelma piirtää toisenkin suorakaiteen ensimmäisen sisälle. Näppäile:

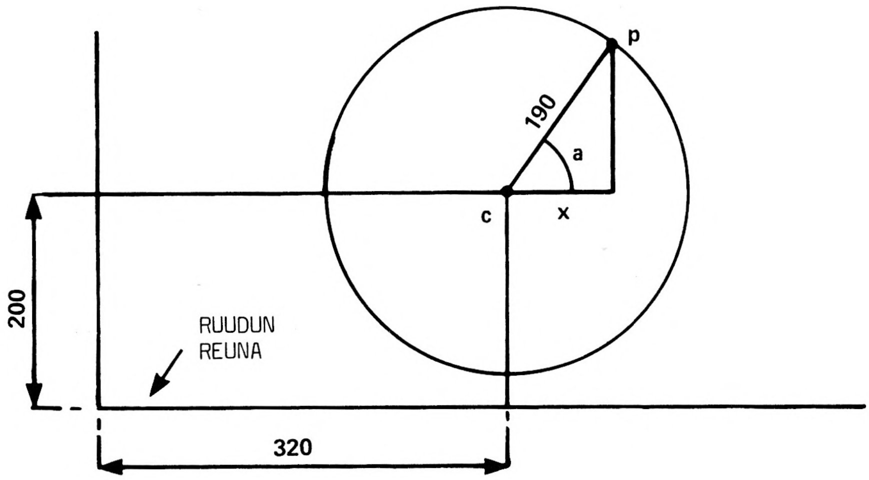
```
60 plot 20,20
70 draw 20,380
80 draw 620,380
90 draw 620,20
100 draw 20,20
200 goto 10
run
```

Katkaise ohjelma lopuksi painamalla [ESC]-näppäintä kahdesti.

YMPYRÄT

Ympyröitä voidaan muodostaa käyttämällä joko plot- tai draw-käskyjä. Eräs tapa ympyrän muodostamiseksi on käyttää plot-käskyä jonka avulla määritellään jokaisen ympyrän kehän pisteen X- ja Y-koordinaatit. Alla olevasta kuvasta käy selville, että ympyrän kehän piste p voidaan määritellä plot-käskyllä käyttämällä X- ja Y-koordinaatteja. Koordinaatit ovat:

x = 190*cos(a) ja y = 190*sin(a)



NEW

Olemme alkaneet käyttää avainsanaa **new** ennen varsinaisen ohjelman näppäilyä. Käsky **new** on määräys tietokoneelle, että sen on nollattava muisti aivan kokonaan samalla tavalla kuin käytettäessä [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä. Erona on kuitenkin se, ettei ruutu tyhjene new-käskyllä, vaan ainoastaan muisti tyhjenee. Voit käyttää tätä hyväksesi silloin, kun haluat säilyttää vanhan ohjelman kuvuruudussa ja katsot siitä joitakin tietoja kirjoittaessasi uuttaa ohjelmaa.

Palatkaamme nyt ympyrän kehän piirtämistehtävään plot-käskyä käyttäen.

Edellisessä ohjelmassa määrittelimme pisteiden sijainnin plot-käskyllä kuvaruudun vasempaan alakulmaan verrattuna. Jos haluamme sijoittaa ympyrän keskelle kuvaruutua, on meidän ensiksi määriteltävä ympyrän keskipisteen koordinaatit 320,200 plot-käskyllä ja sijoitettava sen jälkeen kaikki ympyrän pisteet keskipisteeseen nähden kasvattamalla keskipisteen koordinaatteja tietyssä suhteessa.

Näppäile koneeseen seuraava ohjelma, joka piirtää ympyrän:

```
new
5 cls
10 for a=1 to 360
15 deg
20 plot 320,200
30 plot 320+190*cos(a),200+190*sin(a)
40 next
run
```

Ympyrän sädettä voidaan lyhentää pienentämällä lukua 190 (190 viittaa kuvaalkioihin).

Ympyrän voi piirtää plot-käskyä käyttäen myös eri tavalla (radiaaneina). Kokeile tätä poistamalla rivi 15 ohjelmasta näppäilemällä **15** (ja muista painaa [ENTER]-näppäintä).

Seuraavaksi piirrämme kokonaisen ympyrän, joka muodostuu keskipisteestä lähtevistä säteistä. Tämä tapahtuu korjaamalla rivi 30 siten, että sana **plot** vaihdetaan sanaksi **draw** . Rivi 30 on siis korjauksen jälkeen seuraavan näköinen:

```
30 draw 320+190*cos(a),200+190*sin(a)
```

Kokeile nyt ohjelmaa rivin 15 kanssa ja ilman sitä.

Ehkä huomasitkin, että ohjelmarivillä 40 oli käytetty muotoa **next** muodon **next a** sijasta.

Ohjelmissa tosiaankin voi käyttää muotoa **next** . Tällöin tietokone ratkaisee itse, mihin for-lauseeseen **next** liittyy. Jos ohjelmassa on useampia for- ja next-käskyillä muodostettuja silmukoita, on ohjelman tutkiminen jälkeenpäin varmaankin helpompaa, jos next-käskyn perään liitetään kulloinkin käytettävän muuttujan nimi.

ORIGIN

Edellisessä ohjelmassa käytettiin plot-käskyä ympyrän keskipisteen määrittelemiseksi, jonka jälkeen X- ja Y-koordinaatit lisättiin ympyrän keskipisteeseen. Voimme käyttää myös origin-käskyä sen sijaan, että lisäisimme koordinaattien arvot plot-käskyllä määriteltyyn keskipisteeseen. Ympyrän keskipiste voidaan siis määritellä origin-käskyllä, jonka jälkeen kaikkien ympyrän kehän pisteiden X- ja Y-koordinaatit sijoitetaan ympyrän kehälle (yhden asteen välein) alkupisteestä lähtien. Tämä selviää parhaiten, kun näppäilet seuraavan ohjelman:

```
new
5 cls
10 for a = 1 to 360
15 deg
20 origin 320,200
30 plot 190*cos(a),190*sin(a)
40 next
run
```

Voit kokeilla myös tätä ohjelmaa rivin 15 kanssa ja ilman sitä. Samoin voit vaihtaa rivillä 30 olevan plot-käskyn draw-käskyksi, jolloin saat muodostumaan kokonaisen ympyrän.

Seuraavaksi voit kokeilla ohjelmaa, joka piirtää neljä pienempää ympyrää kuvaruutuun. Näppäile siis:

```
new
5 cls
10 for a=1 to 360
15 deg
20 origin 196,282
30 plot 50*cos(a),50*sin(a)
40 origin 442,282
50 plot 50*cos(a),50*sin(a)
60 origin 196,116
70 plot 50*cos(a),50*sin(a)
80 origin 442,116
90 plot 50*cos(a),50*sin(a)
100 next
run
```

Voit muuttaa myös tätä ohjelmaa poistamalla rivin 15 ja vaihtamalla plot-käskyn draw-käskyksi joko yhdellä tai kaikilla riveillä 30, 50, 70 ja 90.

GOSUB RETURN

Jos jossakin ohjelmassa on joukko käskyjä, jotka ohjelman on suoritettava useampaan kertaan, voidaan nämä käskyt näppäillä aliohjelmaksi, jonka jälkeen käskyt saadaan käyttöön gosub-käskyllä jota seuraa rivinumero.

Gosub-käskyllä toteutetun rutiinin loppukohta merkitään näppäilemällä siihen return-käsky. Tästä kohdasta tietokone palaa sille riville, joka on seuraavana juuri suoritetun gosub-käskyn jälkeen.

Edellisessä ohjelmassa toistui käskyrivi **plot 50*cos(a),50*sin(a)** neljä kertaa. Tämä käskyrivi voidaan näppäillä aliohjelmaksi, jota sitten kutsutaan joka kerta gosub-käskyllä, kun kyseistä ohjelmariviä tarvitaan. Näppäile siis seuraava ohjelma:

```
new
5 cls
10 for a=1 to 360
15 deg
20 origin 196,282
30 gosub 120
40 origin 442,282
50 gosub 120
60 origin 196,116
70 gosub 120
80 origin 442,116
90 gosub 120
100 next
110 end
120 plot 50*cos(a),50*sin(a)
130 return
run
```

Huomasit varmaan, että rivillä 110 on käytetty käskyä **end** , sillä muuten ohjelma tietenkin jatkuisi rivillä 100 olevan käskyn jälkeen ja tietokone suorittaisi rivin 120. Rivi 120 on juuri se rivi, jota kutsumme tarvittaessa gosub-käskyllä.

Lopetamme tämän aihepiirin käsittelyn ohjelmalla, jossa on runsaasti sellaisia ohjelmointikäskyjä ja avainsanoja, jotka pitäisi jo olla sinulle tuttuja. Näppäile siis:

```
new
10 mode 0:border 6:paper 0:ink 0,0
20 gosub 160:for x=1 to 19:locate x,3
30 pen 15:print" ";chr$(238)
40 for t=1 to 50:next t:sound 2,(x+100)
50 next x:gosub 160:for b=3 to 22
60 locate 20,b:pen 7:print chr$(252)
70 cls:gosub 160:next b
80 sound 2,0,100,15,0,0,1
90 gosub 160:border 16,24:locate 20,25
100 pen 14:print chr$(253);
110 for t = 1 to 1000:next t
120 border 6:gosub 160:for f=3 to 24
130 locate 10,(25-f):pen 2
140 print chr$(144):cls:gosub 160
150 sound 7,(100-f),5:next f:goto 10
160 locate 10,25:pen 12
170 print chr$(239):return
run
```

ÄÄNET

CPC 464-tietokoneessa on kaiutin, jonka kautta tietokoneen muodostamat äänitehosteet kuuluvat. Jos laitteistosi kokoonpanoon kuuluu modulaattori/virtalähdeyksikkö MP1 ja tavallinen televisio, säädä television äänenvoimakkuus aivan hiljaiselle.

CPC 464:n oman kaiuttimen kautta kuuluvan äänen voimakkuutta voidaan säätää tietokoneen oikeassa päädyssä olevalla äänenvoimakkuuden säätimellä (VOLUME). Ääni voidaan myös syöttää stereolaitteistosi tuloliitäntään, jossa voi olla esim. merkintä "AUX". Kytkentä suoritetaan tietokoneen takapaneelin vasemmassa reunassa olevasta syöttö/tulostusliitännästä (I/O). Tämä suo sinulle mahdollisuuden kuunnella tietokoneen muodostamaa ääntä HiFi-kaiuttimilla tai kuulokkeilla.

CPC 464:n äänet muodostetaan sound-käskyllä, ja siinä on seitsemän parametriä (eli numeroarvoa, jotka kukin vaikuttavat äänen muodostumiseen). Kahta ensimmäistä parametriä on käytettävä aina, mutta loput voi jättää pois tai valita käyttötilanteen mukaan. Sound-käsky koostuu seuraavista osista:

sound kanavan tila, säveljakso, kesto, voimakkuus, voimakkuusvaippa, sävelvaippa, kohinajakso.

Seuraavissa esimerkeissä näppäillään kanavan tilaksi 1 - toisin sanoen kanavan viitenumero.

Säveljakso

Keski-c:n säveljakso on 478, kuten liitteestä VII ilmenee. Näppäile:

```
new
10 sound 1,478
run
```

Kuulet nyt lyhyen sävelen, joka on 0,2 sekuntia kestävä keski-c.

Kesto

Äänen kesto on 0,2 sekuntia, ellei sitä erikseen määritellä sound-käskyssä. Keston yksikkö on 0,01 sekuntia. Yhden sekunnin mittainen sävel saadaan aikaan käyttämällä 100 yksikköä, ja vastaavasti kahden sekunnin mittainen sävel saadaan aikaan käyttämällä 200 yksikköä. Näppäile:

```
10 sound 1,478,200
run
```

Kaiuttimesta kuuluu nyt keski-c, joka kestää kaksi sekuntia.

Voimakkuus

Tämä numero määrittelee sävelen alkuvoimakkuuden. Numero voi olla väliltä 0 - 7. Jos sound-käskyssä määritellään myös voimakkuusvaippa, on voimakkuuden määrittelyyn käytettävissä numerot välillä 0 - 15. Tietokone käyttää oletusarvona voimakkuudelle numeroa 4, ellei sitä erikseen määritellä. Näppäile:

```
10 sound 1,478,200,3
run
```

Paina mieleesi, millä voimakkuudella tämä ääni kuuluu. Näppäile nyt sama käsky käyttäen suurempaa voimakkuutta kuvaavaa numeroa:

```
10 sound 1,478,200,7
run
```

Ja kuten huomaat, ääni kuuluu nyt paljon voimakkaammin.

Voimakkuusvaippa

Voimakkuusvaipan määrittelemiseen käytetään käskyä **env** . Tässä käskyssä on yleensä neljä parametriä: kolme viimeistä parametriä esiintyvät kaikissa vaipan osissa, joita on käytettävissä enintään viisi. Käytämme niistä nyt vain kolmea, ja käsittelemme asiaa laajemmin luvussa 6. Käskyn muoto on:

env vaipan numero, portaiden määrä, portaan amplitudi (koko), portaan kesto.

Vaipan numero

Tämä on tietylle vaipalle annettava numero, jotta vaippa pystytään määrittelemään sound-käskyssä. Vaipan numero voi olla jokin numeroista 0 - 15.

Portaiden määrä

Tätä käytetään yhdessä portaan keston kanssa. Esimerkki: haluat käyttää kymmentä porrasta, joista kukin on yhden sekunnin mittainen. Tällaisessa tapauksessa portaiden määrä on siis 10.
Portaiden määrän lukuarvo voi olla jokin numero väliltä 0 - 127.

Portaiden amplitudi eli koko

Jokaisen portaan amplitudi voi vaihdella välillä 0 - 15 viimeiseen portaaseen verrattuna. Kaikki 15 voimakkuustasoa ovat samat kuin ne, joita käytetään sound-käskyssäkin. Porrasta voidaan kuitenkin säätää välillä -128...+127, joten voit muuttaa amplitudia haluamaasi suuntaan ylös tai alas, mutta sen lisäksi voit käyttää suurempia lukuja kuin 15, joilla saat aikaan erikoistehosteita. Portaan amplitudia kuvaavan luvun on oltava väliltä -128...+127.

Portaan kesto

Tämä luku määrittelee portaiden välisen ajan sadasosasekunneissa (1 yksikkö = 1/100 sekuntia). Portaan kestoa kuvaavan luvun on oltava väliltä 0 - 255. Näin siis kahden portaan välinen aika voi olla enintään 2,55 sekuntia.

Voimakkuusvaipan kokeilu käy parhaiten, kun näppäilet koneeseen seuraavan ohjelman:

```
5 env 1,10,1,100
10 sound 1,284,1000,1,1
run
```

Rivillä 10 on määritelty ääni, jonka säveljakso on 284 (kansainvälinen a), jonka kesto on 10 sekuntia, jonka alkuvoimakkuus on 1, ja jossa käytetään voimakkuusvaippaa numero 1. Voimakkuusvaippa on puolestaan määritelty rivillä 5 sellaiseksi, että se muodostuu kymmenestä portaasta, ja jokaisen portaan voimakkuus kasvaa arvolla 1 yhden sekunnin välein (100 x 0,01 sekuntia).

Muuta riviä 5 seuraavien esimerkkien mukaisesti ja aja ohjelma (run-käskyllä) joka kerta jotta kuulet, kuinka vaipan muuttaminen vaikuttaa.

```
5 env 1,100,1,10
5 env 1,100,2,10
5 env 1,100,4,10
5 env 1,50,20,20
5 env 1,50,2,20
5 env 1,50,15,30
```

Kokeile lopuksi vielä seuraavaa riviä:

```
5 env 1,50,2,10
```

Kuten huomasit, oli äänen voimakkuus vakio viimeisessä harjoituksessa puolet äänen kestoajasta. Tämä johtuu siitä, että portaiden määrä oli 50 ja kunkin portaan välinen aika oli 0,1 sekuntia. Näin ollen sen ajan pituus, jona amplitudia vaihdeltiin, oli vain viisi sekuntia, mutta sound-käskyllä rivillä 10 määritellyn äänen kesto oli 10 sekuntia (luku 1000).

Suorita nyt omia kokeiluja vaihtelemalla env-käskyn lukuarvoja jotta näet, minkälaisia ääniä saat aikaan.

Sävelvaippa

Sävelvaipan käsky on **ent** . Siinä on yleensä neljä parametriä. Kolme viimeistä parametriä esiintyvät kaikissa käytettävissä olevissa vaipan osissa, joita on enintään viisi. Käytämme niistä nyt vain yhtä, ja käsittelemme asiaa laajemmin luvussa 6. Käskyn muoto on seuraava:

ent vaipan numero, portaiden määrä, portaan säveljakso, portaan kesto.

Vaipan numero

Tämä on tietylle vaipalle annettava numero, jotta vaippa voidaan määritellä sound-käskyssä. Vaipan numeron on oltava väliltä 1 - 15.

Portaiden määrä

Tätä käytetään yhdessä portaan keston kanssa. Haluat ehkä käyttää esimerkiksi kymmentä porrasta, joista kukin kestää yhden sekunnin. Portaiden määrää kuvaavan luvun on oltava väliltä 0 - 239.

Portaan säveljakso

Kunkin portaan säveljakso voi vaihdella välillä -128...+127. Negatiiviset portaat kasvattavat sävelten taajuutta (jolloin sävel soi korkeammalta). Lyhin säveljakso on 0. Tämä täytyy muistaa laskettaessa sävelvaippaa. Säveljaksojen koko alue on esitetty liitteessä VII. Portaan säveljaksoa kuvaava luku voi olla väliltä -128...+127.

Portaan kesto

Tämä luku määrittelee portaiden välisen ajan sadasosasekunneissa (1 yksikkö = 1/100 sekuntia). Portaan kestoa kuvaavan luvun on oltava väliltä 0 - 255. Näin siis kahden portaan välinen aika voi olla enintään 2,55 sekuntia.

Sävelvaipan kokeilu käy parhaiten, kun näppäilet koneeseen seuraavan ohjelman:

```
5 ent 1,100,2,2
10 sound 1,284,200,15,0,1
run
```

Rivillä 10 on määritelty ääni, jonka säveljakso on 284 (kansainvälinen a), joka kestää kaksi sekuntia, jonka alkuvoimakkuus on 15 (maksimi), jossa ei ole voimakkuusvaippaa (määritelty luvulla 0), ja jossa on sävelvaippa numero 1.

Rivillä 5 on määritelty sävelvaippa numero 1, joka koostuu sadasta portaasta, jossa säveljakso kasvaa (taajuus laskee) kahdella 0,02 sekunnin välein (kahden sadasosasekunnin välein).

Muuta nyt rivin 5 käskyä seuraavilla tavoilla ja aja ohjelma joka kerta (run-käskyllä) jotta kuulet, kuinka sävelvaipan muuttaminen vaikuttaa.

```
5 ent 1,100,-2,2
5 ent 1,10,4,20
5 ent 1,10,-4,20
```

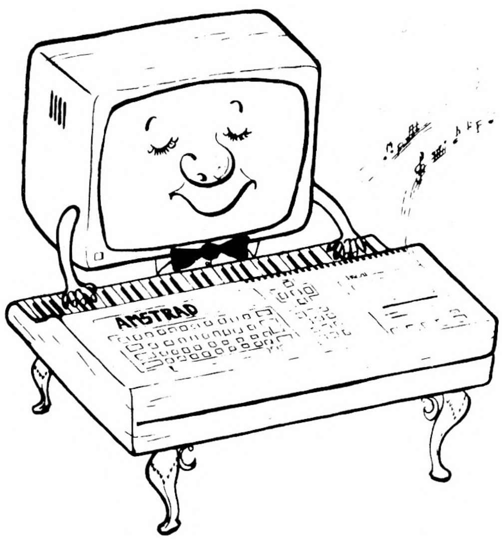
Vaihda sound-käsky ja sävelvaippa näppäilemällä seuraavat rivit:

```
5 ent 1,2,17,70
10 sound 1,142,140,15,0,1
15 goto 5
run
```

Katkaise ohjelma painamalla [ESC]-näppäintä kaksi kertaa.

Nyt osaat jo muodostaa erilaisia ääniä yhdistämällä voimakkuusvaipan, sävelvaipan ja sound-käskyn toisiinsa. Kokeile aluksi seuraavaa ohjelmaa näppäilemällä:

```
new
5 env 1,100,1,3
10 ent 1,100,5,3
20 sound 1,284,300,1,1,1
run
```



Vaihda sitten rivi 10 näppäilemällä:

```
10 ent 1,100,-2,3
```

Vaihda nyt kaikki rivit näppäilemällä:

```
5 env 1,100,2,2
10 ent 1,100,-2,2
20 sound 1,284,200,1,1,1
run
```

Kokeile vielä itse omia muunnelmiasi.

Kohina

Ääntä muodostettaessa voidaan käyttää apuna myös kohinaa, joka lisätään sound-käskyn loppuun. Käytettävissä oleva kohina-alue on 1 - 31. Kokeile tätä lisäämällä sound-käskyn loppuun kohinaa tarkoittava luku, mutta käytä myös voimakkuusvaippakäskyä **env** .

Vaihda edellä olevan ohjelman rivit 5 ja 20 näppäilemällä ne seuraavaan muotoon:

```
5 env 1,100,3,1
20 sound 1,200,100,1,1,1,5
run
```

Yritä taas muodostaa aivan uudenlaisia ääniä muuttamalla voimakkuusvaippaa ja sound-käskyä kohinan kanssa ja ilman kohinaa.

JOHDANTO

Tämä luku on tarkoitettu niille, jotka ovat hypänneet aloittelijoille tarkoitetun peruskurssin yli. Luku sisältää yksityiskohtaiset tiedot laitteiden kytkemisestä, käynnistämisestä sekä näppäimistöön tutustumisesta. Jos tässä luvussa käytetty terminologia ei tunnu tutulta, tutustu ensin käsikirjan alussa olevaan aloittelijoille tarkoitettuun peruskurssiin.

Luvussa käsitellään siis seuraavia asioita:

* Käsikirjassa käytettävää terminologiaa ja kirjoitustapaa.
* Laitteiston käyttökuntoon panoa ja käynnistämistä.
* Näppäimistöön tutustumista.

Muista noudattaa tässä luvussa annettuja ohjeita siitä riippumatta, kuinka paljon olet ollut tekemisissä tietokoneiden ja ohjelmoinnin kanssa. Jos olet juuri avannut pakkauksen ja haluat päästä aloittamaan kokeilut uudella tietokoneellasi, saat tästä luvusta kaiken tarvitsemasi tiedon, jotta pääset aloittamaan kokeilut BASIC-kielellä. Tämä luku on tarkoitettu sellaisille käyttäjille, joilla on jo ennestään jonkin verran kokemusta tietokoneista. Jos sinulla ei ole aikaisempaa kokemusta tietokoneista, sinun kannattaa aloittaa aivan käsikirjan alusta.

TÄRKEÄÄ - tämä sinun on luettava:

TERMINOLOGIA

Käsikirjassa käytetään tästä eteenpäin seuraavia kirjoitustapoja, jotta näppäimistön eri näppäimet ja tekstit, jotka muodostavat osan ohjelmalistausta, erottuisivat selvästi muusta tekstistä:

[ENTER]: Sellaisten näppäinten nimet on merkitty hakasulkeilla [ ], joiden painaminen ei tulosta ruutuun vastaavaa merkkiä.

QWERTYUIOP: Näppäimet, joita painamalla ruutuun tulostuu vastaava merkki, on esitetty tässä muodossa ilman hakasulkeita.

```
10 FOR N = 1 to 1000
```

Yllä oleva esitystapa tarkoittaa sitä, että teksti tulostuu ruutuun tässä muodossa tai teksti näppäillään näppäimistöltä tietokoneeseen. Muista tehdä ero nollan 0 (joka näkyy ruudussa muodossa Ø) ja ison kirjaimen O välillä.

Oletamme, että painat jokaisen ohjelman lopussa tai suoraan käskyrivin lopussa [ENTER]-näppäintä, eikä sinua enää muistuteta tästä seikasta tästä eteenpäin käsikirjan loppuosassa.

Oletamme myös, että näppäilet RUN-käskyn jokaisen syöttämäsi ohjelman päätteeksi. AMSTRADin BASIC muuntaa kaikki pienillä kirjaimilla näppäillyt avainsanat suuriksi kirjaimiksi, kun ohjelma listataan LIST-käskyllä. Tästä eteenpäin olevissa esimerkeissä käytetään isoja kirjaimia, koska avainsanat esiintyvät juuri tässä muodossa ohjelmassa, kun ohjelma listataan. Jos syötät käskyt käyttäen pieniä kirjaimia, huomaat helpommin kirjoitusvirheet, koska väärin näppäillyt avainsanat näkyvät edelleen kuvaruudussa pienillä kirjaimilla, kun ohjelma listataan.

1.1.1 PURETAANPA PAKKAUKSET!

1.1.1.1 Värimonitori

TÄRKEÄÄ!

Käsikirjan alussa on tarkemmat kytkentäohjeet ja kuvat. Kytke laitteistosi sieltä löytyvien ohjeiden mukaan.

Kun olet kytkenyt tietokonelaitteistosi käsikirjan alussa olevan kuvan 1 osoittamalla tavalla, kytke virta monitoriin sen etuosassa olevalla virtakytkimellä ja sitten tietokoneeseen sen oikeassa päädyssä olevalla virtakytkimellä. Monitoriin ilmestyy alla oleva kuva noin 30 sekunnin lämpenemisajan kuluttua:

```
Amstrad 64K Microcomputer  (v1)

©1984 Amstrad Consumer Electronics plc
          and Locomotive Software Ltd.

BASIC 1.0

Ready
█
```

Tätä viestiä voidaan kutsua vaikkapa "nollaus"-, "alku"- tai "herätys"-viestiksi, ja se osoittaa, että tietokone on nollattu aivan kokonaan alkutilaansa. Alkutila on tila, joka vallitsee silloin, kun virta kytketään tietokoneeseen tai kun tietokone nollataan aivan kokonaan näppäimistöltä painamalla tiettyjä kolmea näppäintä oikeassa järjestyksessä samanaikaisesti.

(Nämä näppäimet ovat [CTRL]-, [SHIFT]- ja [ESC]-näppäimet. Kun niitä painetaan tässä järjestyksessä ja kaikkia kolmea näppäintä pidetään yhtäaikaa alhaalla, tapahtuu täydellinen nollaus. Kokeile tätä ennen kuin syötät tietokoneeseen käskyjä).

Voit säätää monitorin kuvan kirkkautta monitorin oikealla puolella olevalla kirkkaussäätimellä (BRIGHTNESS). Näytössä näkyvän kuvan väri on valittu valmiiksi tehtaalla. Jos haluat vaihtaa kuvan väriä (keltaiset kirjaimet sinisellä pohjalla), käy se kylläkin päinsä ohjelmakäskyillä, mutta sinun on hypättävä lukuun 8, jossa käsitellään grafiikan alkeisiin liittyviä BASIC-avainsanoja. Ellet halua hypätä käsikirjassa hieman eteenpäin, on seuraavassa sinulle lyhyt ohjelma, jolla saat aikaan erään parhaimmista väriyhdistelmistä, jossa merkkien luettavuus on erinomainen. Siinä käytetään suuren erottelukyvyn omaavaa 80 sarakkeen näyttömuotoa.

Näppäile:

```
10 MODE 2
20 INK 1,0
30 INK 0,13
40 BORDER 13
```

Voit myös näppäillä yllä olevat käskyrivit suoraan yksittäisinä käskyinä (ilman rivinumeroita). (Säädä kuvan kirkkaus sellaiseksi, että teksti näkyy hyvin ja kuvaa on miellyttävä katsella).

Jos yllä oleva ohjelma ei sano sinulle mitään tai sen oikein näppäily tuottaa vaikeuksia, on sinun palattava takaisin käsikirjan alussa olevaan lukuun "PERUSKURSSI ALOITTELIJOILLE". Tulet huomaamaan, että 80 sarakkeen tekstinäyttö on useimmissa tapauksissa käyttökelpoisin näyttömuoto ohjelmien kehittelyyn. Mielesi tekee ehkä tallettaa yllä oleva lyhyt ohjelma (luettuasi ensin luvun 2) tyhjän kasetin alkuun, jolloin sinun ei tarvitse näppäillä sitä joka kerta kytkiessäsi virran laitteistoon.

HUOM!
Värimonitorin suuri kirkkaus merkitsee sitä, että sinun on tarkkaan varottava rasittamasta silmiäsi istumalla aivan monitorin lähellä tai säätämällä kuva liian kirkkaaksi ympäristön valaistusolosuhteisiin nähden. Katkaise virta ja tee jotakin muuta heti kun tunnet ensimmäiset oireet silmiesi rasittumisesta. Seuraavassa on joitakin neuvoja, joilla voit välttää silmien rasittumista:

1. Kun työskentelet mikrotietokoneen kanssa, pidä huoneessa aina muuta valaistusta niin paljon, että voit lukea tämän käsikirjan tekstiä vaivatta. Käsikirjaa ei missään tapauksessa saa lukea pelkän kuvaputkesta tulevan valon turvin!

2. Säädä kuvan kirkkaus aina minimiin mutta kuitenkin siten, että näet vaivatta sen mitä olet parhaillaan tekemässä.

3. Istu mahdollisimman kaukana kuvaruudusta.

Silmäsi eivät rasitu niin helposti, jos sijoitat pöytälampun monitorin sivulle. Muista kuitenkin sijoittaa lamppu kuvaruudun etureunan takapuolelle, ettei lampun valo heijastu kuvaruudusta.

1.1.2 Vihreä/musta-monitori GT64

Monitorissa GT64 on kolme säädintä edessä näytön alla. Voit säätää säätimillä kuvan kirkkautta (BRIGHTNESS), sävykkyyttä (CONTRAST) eli kuvan kirkkaimpien ja tummimpien osien sävyeroa sekä pystypitoa (V-HOLD), jolla kuva saadaan pysymään vakaasti paikallaan (estetään kuvan "pyöriminen" pystysuunnassa).

Kun olet kerran säätänyt pystypidon säädön kohdalleen, et sitä enää juuri myöhemmin tarvitse. Sen sijaan kirkkaus- ja sävykkyyssäätimiä on aina silloin tällöin tarpeen säätää, jotta katseltava kuva ja huoneen katseluolosuhteet olisivat keskenään sopusoinnussa.

GT64 on yksivärinen monitori (merkit ovat aina saman värisiä, mutta niiden kirkkautta muutetaan haluttujen sävykkyyserojen muodostamiseksi näyttöön). Kun virta kytketään monitoriin, ilmestyy kuvaruutuun sama alkuviesti kuin värimonitorissakin (katso sivua 46) sillä erotuksella, että teksti näkyy kirkkaanvihreänä tummanvihreällä pohjalla.

Vaikka silmäsi voivatkin rasittua liiallisesta puuhailusta monitorin GT64 ja tietokoneen parissa, on yksivärisen monitorikuvan kanssa paljon helpompi työskennellä pitkiäkin aikoja, koska kuva on "pehmeämpi". Saat kaiken hyödyn erityisesti täysleveästä "80 sarakkeen" tekstinäytöstä (80 kirjainta tai numeroa yhdellä ainoalla näyttörivillä), koska yksivärisen kuvaputken erottelukyky (kyky näyttää toisiaan lähekkäin olevia pieniä kuvan yksityiskohtia kirkkaasti ja selvästi) on aivan ylivoimainen kaikkiin muihin paitsi kaikkein kalliimpiin värinäyttöihin verrattuna.

Säädä kuvan kirkkaus (BRIGHTNESS) sellaiseksi, että kuva on riittävän kirkas samalla kuitenkin pitäen huolen siitä, etteivät merkkien muodostamisessa käytettävien viivojen yksittäiset pisteet ala "elämään".

Seuraava lyhyt ohjelma asettaa CPC 464-tietokoneesi 80 sarakkeen näyttömuotoon ja esittelee sinulle valikoiman erilaisia näyttömuotoja. Näppäile:

```
10 REM Ohjelma esittelee erilaisia naytt
otapoja
20 FOR n=0 TO 26
30 MODE 2
40 INK 1,n
50 INK 0,(26-n)
55 BORDER n
60 LOCATE 17,12:PRINT "Nayttotapa vaihtu
u painaessasi jotain nappainta"
70 a$=INKEY$
80 IF a$="" GOTO 70
90 NEXT
100 GOTO 20
```

Yllä oleva ohjelma on myös hyvä esimerkki siitä, miten ohjelmalistaukset on esitetty tässä käsikirjassa. Jotkut ohjelmarivit ovat niin pitkiä (yli 40 merkkiä näyttömuodossa 1), että ne "katkeavat" keskeltä ja jatkuvat seuraavalle riville. Ohjelmalistaukset onkin esitetty juuri siinä muodossa, jossa ne näkyvät myös näyttöruudussa (näyttömuodossa 1). Listauksen ulkomuoto on erilainen riippuen kulloinkin käytettävästä näyttömuodosta (joita siis ovat MODE 0, MODE 1 ja MODE 2). Kokeile esim. yllä olevan ohjelman listausta eri näyttömuodoissa.

Ohjelma ei esittele kaikkia mahdollisia harmaan sävyjä (värejä), mutta saat kuitenkin selvän käsityksen siitä, mitä on tarjolla. Kun löydät mieleisesi sävy-yhdistelmän, pysäytä ohjelma painamalla [ESC]-näppäintä kahdesti. Tällöin näyttöruutuun ilmestyy viesti:

```
*Break*
```

Tästä eteenpäin käsikirjassa viitataan useasti erilaisiin seikkoihin, jotka liittyvät nimenomaan värimonitoriin (jonka voi hankkia lisävarusteena). Joskus voi käydä niinkin, että mielenkiintoisia väri- ja grafiikkatehosteita käyttävät ohjelmat eivät näy juuri lainkaan yksivärisessä monitorissa, vaikka ohjelmia suunniteltaessa on kiinnitetty erityistä huomiota siihen, että värit tuottasivat yksivärisessä monitorissa vastaavia harmaan sävyjä (asiaa käsitellään tarkemmin luvussa 5).

Yksivärisen monitorin etu on sen terävämpi kuva, eivätkä silmät väsy kovinkaan helposti sen kuvaa katseltaessa. Tämä on huomattava etu varsinkin ohjelmien kehittelyvaiheessa. Mutta aikasi harrastettuasi tulet varmaankin huomaamaan, että tietokonekärpänen on puraissut sinua pahemman kerran. Siihen "tautiin" ei liene juuri muuta parannuskeinoa kuin hankkia lisävarusteena saatava värimonitori laitteiston täydennykseksi.

1.1.3 MODULAATTORI/VIRTALÄHDEYKSIKKÖ MP1

MP1 on myös eräs lisävaruste jonka haluat ehkä hankkia, jos nykyinen laitteistosi käsittää CPC 464-tietokoneen lisäksi virheä/musta-monitorin GT64. MP1 suo sinulle nimittäin mahdollisuuden kytkeä tietokoneesi tavalliseen väritelevisioon, jolloin pääset nauttimaan CPC 464:n hienoista väriominaisuuksista kokonaisuudessaan.

TÄRKEÄÄ!

> Käsikirjan alussa on tarkemmat kytkentäohjeet ja kuvat. Kytke laitteistosi sieltä löytyvien ohjeiden mukaan.

Modulaattori/virtalähdeyksikkö (MP1) sijoitetaan sopivalle pöydälle tietokoneen oikealle puolelle television ja pistorasian läheisyyteen. Kytke MP1:stä tuleva suurempi 6-napainen DIN-pistoke tietokoneen takana olevaan monitoriliitäntään (jossa on merkintä MONITOR) käsikirjan alussa sivulla 4 olevan kuvan 2 osoittamalla tavalla. Kytke MP1:stä tuleva pienempi tasavirtasyötön pistoke tietokoneen takaosassa olevaan tasavirtasyötön liitäntään (jossa on merkintä 5 V DC). Kytke MP1:stä tuleva antennijohto television antenniliitäntään. Säädä televisiosi äänenvoimakkuuden säädin minimiasentoon. CPC 464:ssä on oma kaiutin, joten TV:stä kuuluva suhina äänenvoimakkuuden maksimiasennossa virran ollessa kytkettynä tietokoneeseen on aivan normaalia. Kytke virta televisioon ja kytke seuraavaksi virta tietokoneeseen sen oikealla puolella olevalla virtakytkimellä (POWER).

Tietokoneen näppäimistön yläpuolella keskivaiheilla olevan punaisen merkkivalon (ON) pitäisi nyt syttyä, ja seuraavaksi sinun onkin viritettävä televisiosi vastaanottamaan tietokoneen lähettämää signaalia.

Jos televisiossasi on painikkeilla toimiva kanavanvalitsin, valitse vapaa kanava painamalla kanavanvalintapainiketta. Säädä tarvittaessa kanavan hienovirityssäädintä television käyttöohjeessa kuvatulla tavalla. Tietokoneen lähettämä signaali löytyy UHF-kanavan 36 paikkeilta. Voit tarkistaa tämän seikan, jos televisiosi kanavanvalitsimessa on numeroitu viritysasteikko. Säädä hienoviritystä tarpeen mukaan, kunnes television kuvaruudussa näkyy seuraavan näköinen kuva:

```
Amstrad 64K Microcomputer  (v1)

©1984 Amstrad Consumer Electronics plc
          and Locomotive Software Ltd.

BASIC 1.0

Ready
█
```

Kanavan viritys on syytä suorittaa tarkasti, jotta kuva näkyy mahdollisimman terävänä. Teksti näkyy kullankeltaisena tummansinisellä pohjalla, vaikkakin kuva näkyy hieman eri tavalla riippuen kunkin television säädöistä.

Jos televisiossasi on käännettävä kanavanvalitsijan nuppi, käännä nuppia kunnes kuvaruudussa näkyy edellisellä sivulla olevassa kuvassa näkyvä teksti ja kuva näkyy täysin vakaana. (Myös tässä kuva löytyy UHF-kanavan 36 paikkeilta).

Videosignaalin laatu kärsii hieman kulkiessaan kaikkien eri vaiheiden läpi, koska ensin se moduloidaan ja sen jälkeen demoduloidaan. Tulos ei voi koskaan olla niin hyvä kuin suoraan kytketyllä videomonitorilla. Niinpä saattaakin käydä niin, että huomaat televisiosi laadusta riippuen, ettei 80 sarakkeen tekstinäyttö (näyttömuoto 2) näy täysin terävästi. Tällaisessa tapauksessa sinun on syytä käyttää näyttömuotoa 1 tekstejä syöttäessäsi aina kun se vain on mahdollista.

1.2 LÄHDETÄÄNPÄ LIIKKEELLE

Laitteistosi on nyt kytketty toimintakuntoon, eli virtalähde on kytketty, videojohto on kytketty jompaan kumpaan monitoriin tai modulaattoriin, ja virta on kytketty laitteistoon. Tietokoneesi odottaa nyt ohjeita.

Ruudussa näkyvä "alkuviesti" eli "herätysviesti" on ainoa valmis teksti, jonka saat näyttöön tarvitsematta ensin näppäillä käskyjä näppäimistöltä.

Jos BASIC-ohjelmointikieli on sinulle jo entuudestaan tuttu, on varsin todennäköistä, että olet jo ehtinyt näppäillä koneeseen lyhyen ohjelman tutustuaksesi koneeseen. AMSTRADin BASIC-kielessä on paljon tuttuja piirteitä, ja opit helposti sen käskyt lukemalla tätä käsikirjaa eteenpäin. Esitämme sinulle aluksi lyhyen ohjelman, jonka syötettyäsi saat näyttöön kaikki tietokoneen muistiin ohjelmoidut valmiit merkit. Sitä kutsutaan merkkivalikoimaksi, ja siihen kuuluvat kaikki numerot, kirjaimet ja muut merkit, jotka saadaan näyttöön näppäimistön eri näppäimiä käyttäen.

Jotkut ohjelman esittämistä merkeistä ovat sellaisia, ettet saa niitä suoraan näyttöön painamalla näppäimistön näppäimiä, vaan sinun on käytettävä seuraavaa lauseketta (jota käsitellään tarkemmin tuonnempana käsikirjassa):

```
PRINT CHR$(<numero>)
```

Tämä johtuu siitä, että jokainen tietokoneen muistiin tallennettava merkki tallennetaan "tavuna" - ja kuten liitteestä II näet, on tavun arvolla 256 erilaista yhdistelmämahdollisuutta. Koska tietokoneen toiminta edellyttää, että jokaista muistiin tallennettavaa merkkiä kohden on käytettävä yksi kokonainen tavu (se on pienin määriteltävissä oleva tietomäärä, jonka CPC 464 hyväksyy - halusimmepa sitä tai emme), voimme aivan yhtä hyvin käyttää hyväksemme kaikki 256 mahdollista yhdistelmää emmekä tyytyä pelkästään noin 96 vakiomerkkiin, jotka useimmat kirjoituskoneet pystyvät tulostamaan, koska tällöin joutuisimme heittämään hukkaan muut 160 mahdollisuutta.

Merkkivalikoiman vakio-osaa kutsutaan vakiomerkistöksi. Se on määritelty kaikkialla tietokonemaailmassa yleisesti käytettävän ASCII-järjestelmän mukaan. Sana ASCII on lyhenne sanoista American Standard Code for Information Interchange. Se on ensisijaisesti järjestelmä, joka takaa, että tiedot lähtevät tietokoneesta toiseen tietokoneeseen sellaisessa muodossa, että vastaanottajakone ymmärtää lähetetyt tiedot. Se onkin lähes ainoa asia, joka tietokonemaailmassa on tosiaankin yleispätevä. Niinpä suosittelemmekin, että tutustut huolellisesti ASCII-standardiin liittyviin seikkoihin. ASCII-standardin mukainen merkkivalikoima on esitetty liitteessä III, josta löytyvät myös CPC 464:n lisämerkkivalikoima numerokoodeineen.

Näyttöön saadaan eräitä muitakin "kirjoittumattomia" merkkejä käyttämällä CONTROL-näppäintä (jonka merkintä näppäimistössä on [CTRL]) ja eräitä muita näppäimistön näppäimiä. Tästä ei vielä kuitenkaan kannata välittää, sillä sattumanvaraisesta kokeilusta saattaa olla enemmän haittaa kuin hyötyä, ennen kuin opit tuntemaan [CTRL]-näppäimen toiminnan.

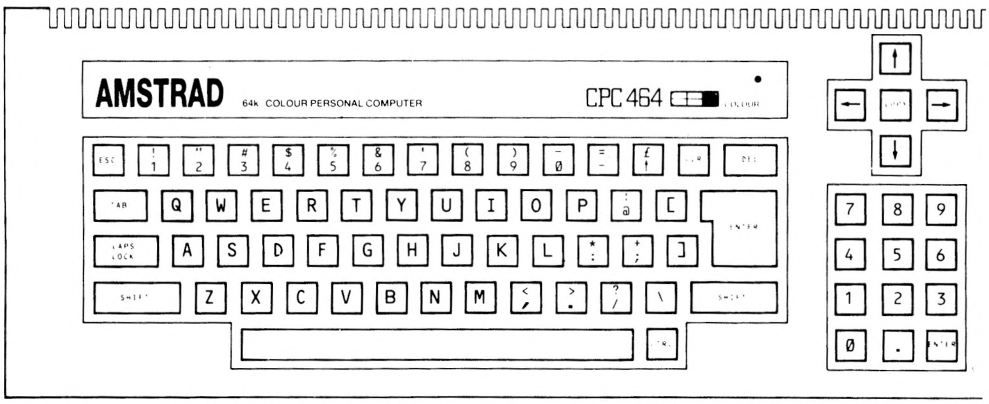
1.2.2

Pääset itse näkemään, miltä kaikki merkit todella näyttävät näppäilemällä koneeseen seuraavalla sivulla olevan ohjelman. Huomaat samalla, miten helppoa CPC 464:n ohjelmointi on. Paina myös mieleesi, että kytkettyäsi virran laitteistoon ja saatuasi näyttöön alkuviestin eli nollausviestin, voit välittömästi aloittaa CPC 464:n ohjelmoinnin muitta mutkitta.

(Jos teet virheen näppäillessäsi ohjelmaa koneeseen, hyppää kohtaan 1.2.7 ja yritä korjata virhe, ettei sinun tarvitse aloittaa ohjelman syöttämistä alusta).

Voit käyttää ohjelmaa koneeseen näppäillessäsi joko pieniä kirjaimia (abc) tai suuria kirjaimia (ABC) (saat isot kirjaimet painamalla [SHIFT]-näppäintä), sillä tietokone hyväksyy ohjelman molemmilla tavoilla näppäiltynä. Sinun TÄYTYY erottaa sanat käyttäen välilyöntejä tai muita erotusmerkkejä (pilkkuja, kaksoispisteitä, tms. tilanteen edellyttämällä tavalla) ohjelmalistauksessa esitetyissä kohdissa, koska AMSTRADin BASIC sallii varattujen sanojen (joista on luettelo liitteessä VIII) käyttämisen muuttujien nimien yhteydessä.

"Fyysinen" näppäimistö näkyy kuvassa 4. Sitä nimitetään "fyysiseksi" näppäimistöksi, koska käyttäjä voi määritellä uudelleen usean näppäimen toiminnan ja muodostaa tällä tavalla koko joukon ylimääräisiä merkkejä. Tätä asiaa käsitellään hieman myöhemmin.



Kuva 4. CPC 464:n näppäimistö

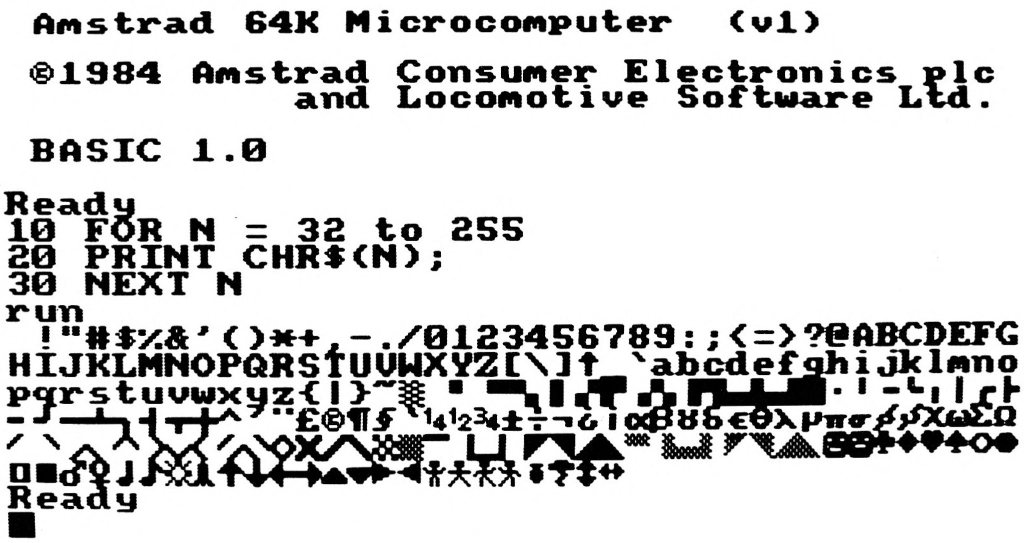
Näppäimistöltä näppäilty käsky tai ohjelmarivi syötetään tietokoneen muistiin painamalla [ENTER]-näppäintä. Samalla se tarkoittaa sitä, että tietokonetta pyydetään käsittelemään kyseisellä rivillä oleva käsky - tai jos rivin alussa on rivinumero, niin tietokonetta pyydetään tallentamaan se muistiin näppäilemääsi ohjelmaan kuuluvana osana.

[ENTER]-näppäimen painamisesta käytetään joissakin yhteyksissä nimitystä "Carriage Return" (vaunun palautus) tai yksinkertaisesti "Return" (palautus). Se juontaa juurensa tietokonepäätteiden varhaisista malleista, jotka perustuivat mekaanisten kirjoituskoneiden periaatteille. Nimitys on edelleen käytössä, ja se on sisällytetty ASCII-merkkivalikoimaan, jossa [ENTER]-näppäimen painallusta vastaava koodi on merkitty kirjaimilla CR. Näppäile koneeseen nyt seuraava ohjelma:

```
10 FOR N = 32 TO 255
20 PRINT CHR$(N);
30 NEXT N
RUN
```

1.2.3

Kuvaruudussa näkyy seuraavaa:



Olet pyytänyt tietokonetta näyttämään koko merkkivalikoimansa juuri kirjoittamallasi lyhyellä ohjelmalla. Jos merkit eivät ilmesty ruutuun, olet tehnyt huomaamatta virheen näppäillessäsi ohjelmaa. Hyppää kohtaan 1.2.7 ja yritä korjata virhe.

Jatkamme siitä oletuksesta, että kaikki on hyvin ja merkit näkyvät ruudussa. Tutkikaamme nyt hieman tarkemmin, miten merkit saatiin ilmestymään ruutuun. Näin opit ymmärtämään, kuinka CPC 464 kommunikoi merkkivalikoimansa välityksellä.

Aivan ensimmäiseksi on huomattava, ettei tietokoneelle annettu käskyä:

```
PRINT "abcdefghijklmn......jne"
```

vaan sille annettiin käsky:

```
PRINT CHR$(N)
```

N on vain kätevä lyhennetty muuttujan merkintätapa. Kirjain on valittu aivan umpimähkään, mutta N sattuu nyt vain olemaan matemaatikkojen suosikki muuttujaa kuvaavana kirjaimena. Muuttuja on puolestaan eräs osa tietokoneen käsittelemään tietoa, joka "muuttuu" ohjelmassa annettujen käskyjen mukaan.

Esimerkiksi numeron 5 tapainen numero on kiinteä (se ei muutu), koska se esiintyy numeroiden 4 ja 6 välissä - näin ollen se ei ole muuttuja. Kirjain N on myöskin kiinteä - se on aakkostoon kuuluva kirjain.

Kuinka tietokone sitten pystyi erottamaan ne toisistaan? Jos olisimme halunneet määritellä kirjaimen N aakkostoon kuuluvaksi kirjaimeksi, olisi N-kirjain pitänyt kirjoittaa lainausmerkkien väliin:

```
"N"
```

Tällöin tietokone olisi tulostanut ruutuun virheilmoituksen **Syntax error** , koska se ei ymmärrä käsky-yhdistelmää FOR "N".

Koska käytimme ohjelmassa N-kirjainta aivan sellaisenaan, ilmoitimme tietokoneelle, että N on muuttuja. BASIC-kielessä on määritelty, että FOR-lausetta on aina seurattava muuttuja. Näin ollen tietokone olettaa, että seuraapa FOR-käskyä mikä hyvänsä, se on juuri muuttuja.

Olemme myös ilmoittaneet tietokoneelle, että N saa arvot välillä 32 - 255 kirjoittamalla ohjelmaan **N = 32 TO 255** . Olemme tällä tavalla määritelleet muuttujan lukualueen, joka alkaa luvusta 32 ja päättyy lukuun 255.

Kun nyt kerran olemme määritelleet muuttujan, on meidän myös ilmoitettava tietokoneelle, mitä sen on tehtävä muuttujalle. Ja seuraavalla rivillä se onkin määritelty:

```
20 PRINT CHR$(N);
```

Se on määräys tietokoneelle, että sen on muunnettava muuttujalle N annettu lukuarvo kyseistä lukua vastaavaksi merkiksi. Tehtävä suoritetaan BASIC-käskyllä **CHR$(N)** . Katsottuaan muistista, mikä merkki vastaa muuttujan N arvoa tietokone tulostaa sen ruutuun.

Rivin lopussa oleva puolipiste aiheuttaa sen, ettei tietokone suorita rivinvaihtoa (eli tulosta seuraavaa merkkiä seuraavan rivin vasempaan reunaan), joka muuten tapahtuisi automaattisesti. Tämä aiheuttaisi sen, että merkit tulostuisivat allekkain ruudun vasemmanpuolimmaiseen sarakkeeseen sen sijaan, että merkit tulostuvat nyt vierekkäin rivi riviltä.

Seuraava rivi ilmoittaa tietokoneelle, että kun se on suorittanut tehtävän lukualueen ensimmäisen luvun (32) kanssa, sen on palattava takaisin sille riville jossa FOR-käsky sijaitsee, ja suoritettava sen jälkeen sama tehtävä uudelleen seuraavan (NEXT = suom. seuraava) arvon kanssa, jonka se antaa muuttujalle N. Tällaista menettelytapaa kutsutaan silmukaksi, ja se on eräs kaikkein tärkeimmistä tietokoneen ohjelmointiin ja käyttöön liittyvistä perusasioista.

Koska FOR-silmukka on eräs tärkeimmistä tietokoneisiin liittyvistä perusasioista, se esiintyy kaikissa ohjelmointikielissä ainakin jossakin muodossa. Ohjelman laatija säästyy pitkiltä ja toistuvilta näppäilyiltä, jotka ilman silmukkaa täytyisi näppäillä käsin. Omaksut varmasti nopeasti silmukan käytön omassa ohjelmoinnissasi.

Kun FOR-silmukka saavuttaa ohjelmassa määritellyn muuttujan lukualueen loppupään (255) toiminta pysähtyy ja tietokone katsoo seuraavaksi, onko rivin 30 jälkeen vielä ohjelmarivejä. Koska rivejä ei enää ole, se yksinkertaisesti pysähtyy ja palaa valmiustilaan tulostamalla näyttöön ilmoituksen **Ready** . Tällä ilmoituksella tietokone kertoo sinulle, että se on valmis ottamaan vastaan uusia käskyjä. Voit myös käskeä tietokonetta suorittamaan ohjelman uudelleen RUN-käskyllä. Ohjelma on visusti tallennettuna tietokoneen muistiin, jossa se pysyy niin kauan kunnes ilmoitat toisin tietokoneelle - tai jos katkaiset virran tietokoneesta, jolloin kaikki tiedot (ohjelmat, muuttujat, jne.) häviävät muistista, ellet talleta niitä kasetille.

Edellä oleva ohjelma on oiva esimerkki eräästä toisestakin tietokoneisiin liittyvästä perusasiasta: kaikki mitä tietokone tekee, liittyy jollain tavalla numeroihin. Tietokone tulosti näyttöön aakkoset - ja koko joukon muita merkkejä - käyttäen numeroa viitteenä tulostettavaksi halutulle merkille. Kun painat esim. A-kirjaimella merkittyä näppäintä, et suinkaan pyydä tietokonetta kirjoittamaan A-kirjainta ruutuun vaan pyydät tietokonetta katsomaan siihen osaan tietokoneen muistia, joka sisältää A-kirjaimen ruutuun kirjoittamiseksi tarvittavat numerotiedot. Näiden tietojen todellinen sijainti määräytyy numerokoodin perusteella, joka aktivoituu painaessasi näppäimistön näppäintä.

Jokaisella merkillä on vastaava numerokoodi, ja koodit löytyvät tämän käsikirjan lopussa olevasta liitteestä III.

Vastaavasti näyttöön tulostuvalla merkillä ei ole mitään tekemistä "kirjoittamisen" kanssa, vaan kaikki liittyy koko ajan numeroihin.

1.2.4

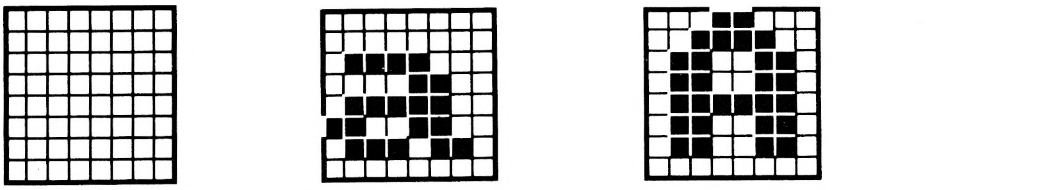
Sinun ei kannata välittää, jos et ymmärrä kaikkia tällä sivulla olevia teknisiä seikkoja ja ammattitermejä. On tärkeää, että selitämme tarkkaan tavan, jolla tietokone käsittelee antamasi käskyt ja tuottaa halutut tulokset. On kuitenkin todennäköistä, että vain teknisesti ajattelevat henkilöt antavat täyden arvon selityksillemme. Jos tämä luku tuntuu sinusta monimutkaiselta, hyppää lukuun 1.2.5.

Esimerkiksi kirjaimen A numerokoodi on 97. Tietokone ei ymmärrä myöskään numerokoodia 97 vaan se on muunnettava ihmisten käyttämästä desimaalimuodosta tietokoneen ymmärtämään muotoon. Tätä koodimuotoa kutsutaan yleisesti konekoodiksi, ja konekoodiin liittyviä periaatteita käsitellään liitteessä II.

Olemme tottuneet käyttämään jokapäiväisessä elämässämme numeroista desimaalimuotoa, ja sen muuntaminen heksadesimaalimuotoon tietokonetta varten saattaa ensinäkemältä näyttää monimutkaiselta. Kymmenjärjestelmään perustuvien lukujen käsittely tuntuu meistä niin luonnolliselta, että jonkun heksadesimaalimuodon ajatteleminen saattaa tuntua samalta kuin yrittäisi syödä ruokaansa veitsi ja haarukka väärissä käsissä.

Heksadesimaalimuodon (HEX) ymmärtäminen vaatii hieman vaivannäköä, mutta kun sen kerran ymmärtää, loksahtavat monet tietokoneisiin liittyvät asiat omille paikoilleen ja numerointijärjestelmän hieno rakenne tuntuukin sen jälkeen selvältä.

Kun tietokone on muuntanut A-näppäimen painalluksen sellaiseksi numeroksi jonka se ymmärtää, se tutkii painallukseen liittyvää tiettyä muistin osaa, josta se löytää erään toisen numerosarjan, joka määrittelee kyseisen merkin. Tämä tarkoittaa siis sitä, että näytössä näkyvä merkki koostuu tietolohkosta, joka on tallennettu muistiin numeromatriisina:



Kuva 5. Tyhjä merkkiruudukko, pieni a ja iso A

Matriisi koostuu pisteriveistä ja -sarakkeista. Merkki saadaan näkyviin sytyttämällä tai sammuttamalla halutut ruudukon pisteet, ja jokainen piste määritellään tietokoneen muistiin tallennetuilla tiedoilla. Jokainen CPC 464:n näytössä näkyvä merkki koostuu solusta, jossa on 8 riviä ja 8 saraketta (8 x 8 matriisi). Ja jos et löydä koneen 225 merkkiä käsittävästä valmiista merkkivalikoimasta tarvitsemaasi merkkiä, voit itse määritellä omia merkkejäsi (esim. ä- ja ö-kirjaimet) käyttäen luvussa 8 käsiteltäviä avainsanoja SYMBOL ja SYMBOL AFTER.

Käyttäjän määrittelemät merkit voivat muodostua minkälaisesta pisteyhdistelmästä hyvänsä, kunhan pisteiden määrä on välillä 0 - 64. Pisteet voivat olla siis minkälaisessa järjestyksessä hyvänsä, joten merkkivalikoima, jossa on käytetty kaikki matriisin pisteiden yhdistelytavat, käsittää paljon laajemman merkkimäärän. Ota lisäksi huomioon, että voit muodostaa merkeistä kokoamalla suurempia merkkilohkoja, eikä käyttäjän määrittelemien merkkien käyttömahdollisuuksia rajoita oikeastaan mikään muu kuin niiden tekemiseen käyttämäsi aika ja oma mielikuvituksesi.

1.2.5 Takaisin ohjelmaan!

Ensimmäisen ohjelman tulos näkyy ruudussa melko epäsiistinä. Ruudun yläosassa näkyy edelleen alkuviesti. Tulos on siistimpi, jos pyyhimme ruudun puhtaaksi ennen kuin käynnistämme ohjelman ajon. Tilanne korjaantuu, kun lisäämme ohjelmaan yhden rivin.

Näppäile seuraava käskyrivi sille riville, jossa kohdistin eli "kursori" on (kohdistimeksi kutsutaan sitä neliötä, joka on heti sanan **Ready** alla aivan vasemmassa reunassa. Ellet tiennyt tätä jo entuudestaan, ei sinun kannata lukea tätä lukua ennen kuin olet ensin lukenut läpi käsikirjan alussa olevan peruskurssin). Näppäile siis:

```
5 CLS
RUN
```

Ja kuten näet, ruutu tyhjenee aivan puhtaaksi ennen kuin merkkivalikoima alkaa tulostua kuvaruudun vasemmasta yläkulmasta.

Tämä on myös hyvä esimerkki BASIC-ohjelmointikieleen liittyvistä perusasioista; nimittäin sillä ei ole mitään väliä, missä järjestyksessä syötät ohjelman rivinumerot - eikä ohjelman edes tarvitse olla esillä näytössä voidaksesi lisätä siihen rivin tai rivejä sen jälkeen kun ohjelma on jo kerran syötetty tietokoneen muistiin.

Tietokone järjestää rivinumerot aina numerojärjestykseen ennen kuin se aloittaa ohjelman suorituksen. Voit tarkistaa tämän LIST-käskyllä.

1.2.6 Ohjelman listaus LIST-käskyllä

Voit helposti tarkistaa, mitä tietokone on tallentanut ohjelmamuistiinsa pyytämällä ohjelmalistausta. Tämä tapahtuu näppäilemällä:

```
LIST
```

Käskyn seurauksena ruudussa näkyy:

```
5 CLS
10 FOR N = 32 TO 255
20 PRINT CHR$(N);
30 NEXT N
Ready
```

Tämä ohjelma pysyy CPC 464:n muistissa, kunnes teet jonkin seuraavista asioista:

- Katkaiset virran tietokoneesta.
- Suoritat nollauksen painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä tässä järjestyksessä ja pitämällä näppäimiä alhaalla, jolloin tietokoneen muisti nollautuu (tyhjenee).
- Lataat (LOAD-käskyllä) tai ajat (RUN-käskyllä) kasetilta jonkin ohjelman.
- Näppäilet **NEW [ENTER]** . Tämä nollaa kaikki muuttujat ja tyhjentää ohjelmamuistin, mutta mm. näyttömuoto ja värit pysyvät muuttumattomina.

Ohjelmoikaamme nyt yksi toimintonäppäin siten, että se suorittaa samat tehtävät kuin jos näppäilisimme **[ENTER] CLS:LIST [ENTER]**. Tämä on eräs CPC 464:n ominaisuuksista, joka nopeuttaa huomattavasti ohjelman syöttö- ja kehitysvaihetta. Tämä tapahtuu näppäilemällä:

```
KEY 138,CHR$(13)+"CLS:LIST"+CHR$(13)
```

Paina nyt numeronäppäimistön (joka on heti kasettinauhurin vasemmalla puolella) desimaalipistettä. Voit esiohjelmoida tällä tavalla jopa 32 näppäintä, ja voit valita mitkä hyvänsä näppäimistön näppäimet uudelleen määriteltäviksi. Huomaa kuitenkin että määritellessäsi jollekin näppäimelle uuden toiminnon, et voi käyttää sitä alkuperäiseen tarkoitukseensa. Asiaa käsitellään luvussa 8 KEY-käskyn kohdalla.

Jos näppäilet koneeseen pitkiä ohjelmia, sinun kannattaa suorittaa seuraava toimintonäppäimen määritys:

```
KEY 138,CHR$(13)+"CLS:LIST"
```

Kun nyt painat samaa desimaalipisteen näppäintä, voit syöttää koneeseen ne rivinumerot, joiden väliltä haluat listauksen (jos ohjelmasi viimeinen rivinumero on esim. 500, voit pyytää listauksen vaikkapa riveistä 350 - 370 näppäilemällä 350-370) (aloita rivinumeroiden syöttö välilyönnillä). Jos painat näppäintä kahdesti, saat koko ohjelman listauksen.

Joskus värejä kokeillessasi saattaa käydä siten, että sekoat väriyhdistelmissä niin, ettet pysty lukemaan näytössä olevaa tekstiä sen vuoksi, että merkkien ja taustan värille on annettu sama arvo. Niinpä sinun kannattaakin suorittaa seuraava toimintonäppäimen määrittely:

```
KEY 139,CHR$(13)+"mode 2:ink 1,0:ink 0,9
"+CHR$(13)
```

Tämän jälkeen sinun tarvitsee vain painaa pienempää [ENTER]-näppäintä (sitä joka on numeronäppäimistön yhteydessä), ja saat ruutuun käyttöösi sellaisen väriyhdistelmän, joka näkyy (kunhan kirkkaussäädin (BRIGHTNESS) on säädetty riittävän kirkkaalle). Ohjelma ei häviä muistista painaessasi näppäintä.

Käyttäjän määrittelemät näppäinkoodit nollautuvat samalla kun tietokonekin nollautuu (katso edellistä sivua), koska näppäinten on oltava taas normaalissa käytössä ohjelmointikäskyjen laadintaan. Kun alat vähitellen päästä selville siitä, mitkä toimintonäppäimet haluat määritellä uudelleen, tee niistä lyhyt ohjelma ja talleta se kasetille, josta saat sen aina kätevästi käyttöösi aloittaessasi työskentelyn.

1.2.7 Korjailun eli editoinnin alkeet

Ohjelmia koneeseen näppäiltäessä syntyy joskus väistämättä virheitä. Niin että tervetuloa vain kaikki ne lukijat, jotka hyppäsivät tähän lukuun luvusta 1.2.2!

CPC 464 on yritetty tehdä virheiden korjaamisen kannalta niin yksinkertaiseksi kuin mahdollista samalla yrittäen välttää niitä ongelmia, joita syntyy kirjoitettaessa uusia merkkejä vahingossa niiden merkkien päälle, joita ei edes ole tarkoitus muuttaa.

Kohdistimen ohjaamiseen kuvaruudussa käytetään omaksi ryhmäkseen koottuja nuolinäppäimiä, jotka ovat heti kasettinauhurin vasemmalla puolella. Kohdistin puolestaan oli se kuvaruudussa näkyvä neliö, joka määrittelee sen kohdan, johon tekstiä seuraavaksi näppäillään. Tietokoneelle ilmoitetaan nuolinäppäimillä se näytön kohta, johon muutos halutaan tehdä.

Kun näppäilet virheen rivinumerolla varustetulla rivillä kuten esimerkiksi:

```
1Ø FOR N = 332 TO 255
```

Sinulla on useampia vaihtoehtoja:

1. Voit painaa [ENTER]-näppäintä ja näppäillä koko rivin uudelleen. Kun painat [ENTER]-näppäintä sen jälkeen kun olet näppäillyt uuden rivin, pyyhkiytyy virheellinen rivi pois muistista ja muistiin tallentuu uusi samalla rivinumerolla alkava virheetön rivi.

2. Voit painaa [←]-näppäintä ja siirtää kohdistimen virheen kohdalle:

```
1Ø FOR N = 332 TO 255
```

Tietokone käyttää käänteistä videota kohdistimen alla olevan merkin esittämiseen. Toisin sanoen normaalisti kohdistimen värinen merkki on muuttunut samanväriseksi kuin taustakin (eli "paperi"), joten se näkyy merkin päälle laitetun kohdistimen läpi.

Kun painat nyt [CLR]-näppäintä, niin kohdistimen kohdalla oleva merkki häviää ja muut kohdistimen oikealla puolella olevat merkit siirtyvät yhden merkkipaikan vasemmalle:

```
1Ø FOR N = 32 TO 255
```

Paina seuraavaksi uudelleen [ENTER]-näppäintä, jolloin juuri korjaamasi rivi tallentuu tietokoneen muistiin. Kohdistinta ei tarvitse siirtää rivin loppuun korjauksen jälkeen, vaan rivi tallentuu kokonaisena muistiin kohdistimen paikasta riippumatta.

3. Voit myös siirtää kohdistimen sen merkin oikealle puolelle, jonka haluat poistaa:

```
10 FOR N = 332 TO 255
```

Kun painat nyt [DEL]-näppäintä, poistuu kohdistimen vasemmalla puolella oleva merkki. Samalla kohdistin vetää perässään loppuriviä vaikuttamatta millään tavalla kohdistimen kohdalla olevaan merkkiin.

Kun painat [ENTER]-näppäintä korjauksen jälkeen, tallentuu uusi rivi muistiin kuten aiemminkin.

```
10 FOR N = 32 TO 255
```

1.2.8

Edellä kuvatut korjausmenetelmät ovat aivan verrattomia silloin, kun huomaat virheen ennen kuin olet päässyt rivin loppuun ja ehtinyt painaa [ENTER]-näppäintä. Useimmat virheet tulevat kuitenkin epähuomiossa, ja ne paljastuvat vasta sitten kun yrität ajaa ohjelmaa, jolloin tietokone tulostaa ruutuun virheilmoituksen (katso liitettä VIII).

Koko joukko virheistä on sellaisia, joiden seurauksena tietokone tulostaa näyttöön virheellisen rivin siten, että korjailukohdistin on valmiiksi laitettu ensimmäiseen merkkipaikkaan (vasempaan reunaan). Tällaisessa tapauksessa voit käyttää edellä kuvattuja korjausmenetelmiä samalla tavalla, kuin jos olisit huomannut virheen ennen rivin syöttämistä muistiin.

Ellei tietokone tulosta ruutuun virheellistä riviä edellä kuvatulla tavalla, sinun on pyydettävä ohjelmalistaus LIST-käskyllä, etsittävä virhe, ja korjattava se.

1.2.9 Korjailu kopiointikohdistimella

Tätä menetelmää käytettäessä pyydetään aluksi ohjelman listaus LIST-käskyllä. (Jatkamme harjoituksia samalla kokeiluohjelmalla, joka mahtuu kerralla ruutuun):

```
5 CLS
10 FOR N = 32 TO 255
20 PRINT CHRS(N);
30 NEXT N
```

Virhe on nyt rivillä 20: siinä on S-kirjain dollarinmerkin $ paikalla ($ ilmoittaa tietokoneelle, että $-merkin jälkeen olevat merkit ovat tekstiä eikä numerotietoa). Voit näppäillä rivin 20 uudelleen alusta loppuun ja syöttää sen muistiin painamalla [ENTER]-näppäintä. Toinen vaihtoehto on se, että käytät korjaamiseen korjailukohdistinta seuraavasti:

Pidä [SHIFT]-näppäin alhaalla (kumpi hyvänsä näppäimistön vasemmalla tai oikealla puolella olevista [SHIFT]-näppäimistä käy yhtä hyvin) ja paina sitten kohdistimen ylöspäinsiirtonäppäintä [↑].

Voit siirtää kohdistinta ylöspäin rivi kerrallaan painalemalla näppäintä, tai voit myös pitää näppäimen alhaalla ja odottaa, kunnes automatiikka siirtää sen puolestasi. Kohdistin pysähtyy samalla hetkellä kun nostat sormesi pois näppäimeltä, ja opit käyttämään tätäkin toimintoa nopeasti pienen harjoittelun jälkeen. Jos kohdistin menee liian ylös, voit siirtää sitä takaisin alaspäin kohdistimen alaspäinsiirtonäppäimellä [↓] pitäen samalla [SHIFT]-näppäimen alhaalla.

Tätä menetelmää käytettäessä varsinaisesta kohdistimesta eroaa "kopiointikohdistin" (molemmat ovat samannäköisiä), ja kopiointikohdistin viedään sille riville, jolla korjaus halutaan tehdä. Kopiointikohdistimen on oltava aluksi korjattavan rivin ensimmäisessä merkkipaikassa (aivan vasemmassa reunassa).

```
5 CLS
10 FOR N = 32 TO 255
20 PRINT CHR$(N);
30 NEXT N
Ready
■
```

Jos olisit siirtänyt varsinaisen kohdistimen korjattavalle riville käyttämällä kohdistimen siirtonäppäimiä painamatta samalla [SHIFT]-näppäintä, ei tietokone olisi hyväksynyt tätä toimenpidettä. Tietokone hyväksyy käskyiksi vain sellaiset merkit, jotka syötetään välittömästi varsinaisen kohdistimen perään.

Jos kuitenkin kokeilet rivin korjaamista tällä tavalla ja kirjoitat uuden rivin vanhan päälle, pääset helposti satimesta painamalla [ESC]-näppäintä <u>ennen kuin</u> painat kumpaakaan [ENTER]-näppäintä tai jotain toimintonäppäintä, johon on ohjelmoitu käsky **chr$(13)** . Jos näppäilet vahingossa NEW-käskyn ja painat sen jälkeen [ENTER]-näppäintä, häviää ohjelmasi muistista lopullisesti - kannattaa siis olla tarkkana!

Kun olet alkanut näppäillä merkkejä riville ja jos yrität siirtyä riviltä pois painamatta [ENTER]-näppäintä, kuuluu tietokoneen kaiuttimesta merkkiääni tullessasi kielletyn alueen reunaan. Pääset pinteestä painamalla [ENTER]-näppäintä, eikä ohjelmasta häviä mitään tietoja - ellet ole näppäillyt aivan ensimmäisiksi merkeiksi jo käytössä olevaa rivinumeroa. Tällöin käy niin, että sen rivin päälle kirjoittuu uusia merkkejä, jonka rivinumeron olet näppäillyt. Silloin ei auta muu kuin näppäillä koko rivi uudelleen.

Kun nyt olet saanut kopiointikohdistimen siirretyksi oikean rivin alkuun, painele COPY-näppäintä kunnes kopiointikohdistin on sen merkin kohdalla, jonka haluat vaihtaa. (Kun totut nopeuteen jolla kopiointikohdistin liikkuu, voit pitää COPY-näppäimen alhaalla, jolloin kopiointikohdistin liikkuu nopeammin).

```
5 CLS
10 FOR N = 32 TO 255
20 PRINT CHRS(N);
30 NEXT N
Ready
20 PRINT CHR■
```

Vapauta nyt [COPY]-näppäin ja näppäile dollarinmerkki $. Se ilmestyy varsinaisen kohdistimen kohdalle, ja samalla kohdistin siirtyy yhden merkkipaikan oikealle tavalliseen tapaan.

```
20 PRINT CHR$■
```

Seuraavaksi sinun täytyy siirtää kopiointikohdistin väärin näppäilemäsi S-kirjaimen ohi. Tämä tapahtuu pitämällä [SHIFT]-näppäin alhaalla ja painamalla kohdistimen oikeallesiirtonäppäintä [→]. Tällöin kopiointikohdistin siirtyy oikeanpuoleisen sulkumerkin (kohdalle. Vapauta [SHIFT]-näppäin ja pidä [COPY]-näppäin alhaalla, kunnes kopiointikohdistin on liikkunut koko rivin yli. Kun painat seuraavaksi [ENTER]-näppäintä, korvaa kuvaruudun alaosassa oleva korjattu rivi listauksessa näkyvän virheellisen rivin.

Voit myös yhdistää kaikki edellä kuvatut korjailutavat kopioimalla koko virheellisen rivin [COPY]-näppäimellä ja käyttämällä varsinaisen kohdistimen kanssa kohdistimen siirtonäppäimiä (nuolinäppäimiä) sekä [CLR]- ja [DEL]-näppäimiä ennen kuin painat [ENTER]-näppäintä rivin lopussa. Tällöin [SHIFT)-näppäintä ei käytetä. Jos painat [CTRL]-näppäintä ja kohdistimen vasemmallesiirtonäppäintä [←] tai kohdistimen oikeallesiirtonäppäintä [→], siirtyy kohdistin vastaavasti joko korjattavan rivin vasempaan tai oikeaan reunaan yhdellä kerralla.

Harjoittele edellä kuvattuja korjaustoimenpiteitä niin huomaat, että korjailu alkaa käydä helpommin pienen harjoittelun jälkeen.

Jäljellä on vielä eräs tapa, jota voit käyttää rivin korjaamiseen. Se tapahtuu EDIT-käskyllä. Näppäile:

```
EDIT 20
```

Tietokone vastaa antamaasi käskyyn seuraavasti:

```
20 PRINT CHRS(N);
```

Tämän jälkeen voit käyttää varsinaisen kohdistimen siirtonäppäimiä yhdessä [CLR]- ja [DEL]-näppäinten kanssa edellä kuvatulla tavalla. Kun olet korjannut rivin, paina [ENTER]-näppäintä kuten ennenkin. Jos jokin menee vikaan ja korjailu jumiutuu, paina [ESC]-näppäintä ja pyydä uusi listaus LIST-käskyllä. Tietokone ei sotke sitä riviä, jolla painat [ESC]-näppäintä.

Pyydä nyt lopuksi vielä kerran uusi listaus LIST-käskyllä. Ruutuun ilmestyy ohjelman korjattu versio. Jos listauksessa on kuitenkin vielä virheitä, korjaa ne uudelleen.

Olemme käsitelleet tähän mennessä pelkästään CPC 464:n oletusarvona valitsemaa näyttömuotoa 1 (mode 1). CPC 464:ssä on kuitenkin kaksi muutakin näyttömuotoa, joten näppäile:

```
MODE 0
RUN
```

Ja kuten varmaan huomasit, näyttömuodon vaihtokäsky tyhjentää ruudun. Seuraavaksi ruutuun tulostuu tietokoneen merkkivalikoima 20 merkkiä leveille riveille:

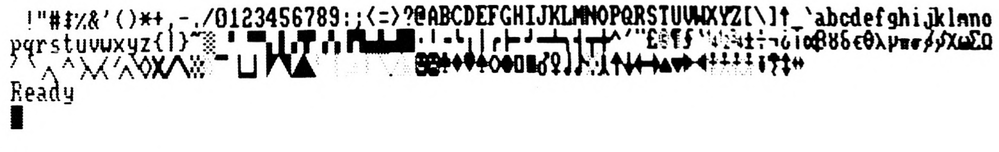


Alkuperäiseen näyttömuotoon palataan näppäilemällä:

```
MODE 1
RUN
```

Olemme nyt jälleen siinä mistä aloitimmekin. Kolmannessa näyttömuodossa on 80 saraketta, ja se saadaan käyttöön näppäilemällä:

```
MODE 2
RUN
```



Näin olemme päässeet alkuun tutkiessamme CPC 464:n ominaisuuksia. Pidemmälle ehtineet käyttäjät alkavat jo varmaan valmistautua muuttamaan suosikkiohjelmiaan CPC 464:n BASIC-murteelle sopiviksi. Vähemmän kokemusta omaavien käyttäjien kannattaa lukea alkeita käsittelevät luvut läpi, koska sieltä saa hyvän käsityksen niistä BASIC-kielen piirteistä, jotka ovat ominaisia juuri CPC 464:lle.

LUKU 2: TIETOKONEEN OMA KASETTINAUHURI

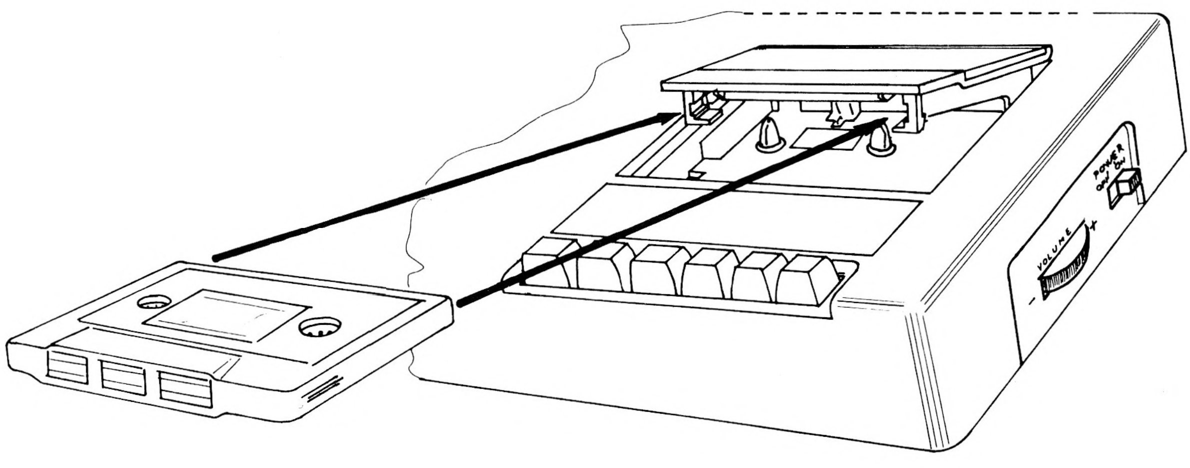
Tässä luvussa käsitellään seuraavia asioita:

- Datakasettien ja äänentoistolaitteissa käytettävien kasettien välisiä yhtäläisyyksiä ja eroavaisuuksia.
- Ohjelmien lataaminen kasetilta tietokoneen muistiin ja ohjelmien ajaminen.
  - Esittelykasetti.
- Valinnaiset tiedonsiirtonopeudet kasettikäytössä.
- Ohjelmien tallettaminen kasetille.
- Lukuvirheet.

Voit tallentaa tietoa CPC 464:n muistiin vain silloin kun virta on kytketty tietokoneeseen. Siksi sanotaankin, että tietokoneen muisti on "haihtuva" tallennusväline. Jos haluat, että ohjelmat tai tiedot pysyvät tallessa myös silloin kun virta on katkaistu, on ohjelmat tai tiedot talletettava kasetille (tai jollekin muulle "haihtumattomalle" tallennusvälineelle, kuten lisävarusteena saatavalle tietolevyjärjestelmälle).

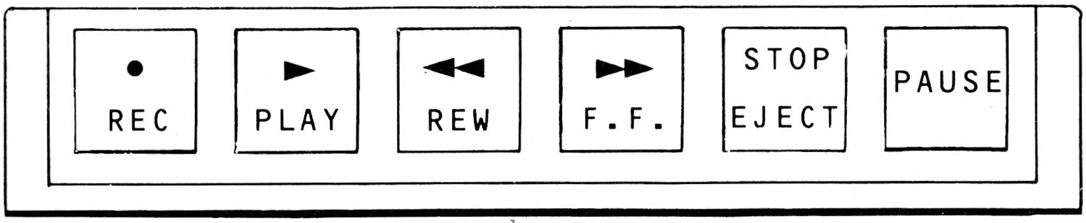
2.1 KASETTINAUHURIN KÄYTTÖSÄÄTIMET

Näppäimistöyksikön oikeassa päädyssä on kuvassa 2.1 näkyvä kasettinauhuri. Tämä kasettinauhuri on mekaniikaltaan lähes samanlainen kuin äänentoistolaitteissa käytettävät kasettinauhuritkin. Erona on vain se, että signaaleja ohjaavien elektroniikkapiirien toiminta on optimoitu erityisesti tietokonekäyttöön.



Kuva 2.1. Kasetin oikea laittotapa kasettinauhuriin

Myös kasettinauhurin käyttöpainikkeet toimivat samalla tavalla kuin useimmissa äänentoistoon käytettävissä kasettinauhureissakin:



Kuva 2.2. CPC 464:n kasettinauhurin käyttöpainikkeet

Kasettinauhurin painikkeita on painettava huomattavasti kovempaa kuin tietokoneen näppäimistön näppäimiä.

Äänityspainike [REC]

Äänityspainiketta käytetään yhtä aikaa toistopainikkeen [PLAY] kanssa äänitettäessä tietoja kasetille. Äänitys käynnistetään ohjelmakäskyllä. Äänityspainike ei painu alas, ellei kasettipesään ole laitettu sellaista kasettia jossa varmuuskielekkeet ovat paikallaan (katso kuvaa 2.3) ja ellei kasettipesän kansi ole kiinni. Kasettinauhuri laitetaan äänitystilaan painamalla [REC]-painiketta - ja samalla kun [REC]-painike pidetään alhaalla, painetaan myös toistopainiketta [PLAY]. Tämän jälkeen tietokone tallettaa tiedot kasetille, kunhan se ensin saa käynnistyskäskyn parhaillaan muistissa olevasta ohjelmasta tai kun käyttäjä näppäilee SAVE-käskyn suoraan näppäimistöltä.

Toistopainike [PLAY]

Toistopainiketta käytetään kasetilla olevien ohjelmien lataamiseen tietokoneen muistiin (LOAD-käskyllä) ja ajamiseen (RUN-käskyllä). Tällöin tietokone lukee tiedot kasetilta saatuaan ensin käskyn muistissa parhaillaan olevasta ohjelmasta tai kun käyttäjä antaa käskyn suoraan näppäimistöltä. Palautusmekanismi vapauttaa [REC]- ja [PLAY]-painikkeet nauhan kelautuessa loppuun.

Takaisinkelauspainike [REW]

Tätä painiketta käytetään kelaamaan nauhaa oikeanpuoleiselta kelalta vasemmanpuoleiselle kelalle. Takaisinkelaustoiminto ei kuulu vapautusmekanismin piiriin, joten takaisinkelausta ei saa jättää käyntiin nauhan loputtua, koska muuten nauhan vetomoottori saattaa vioittua ylikuumetessaan.

Eteenpäinkelauspainike [F.F.]

Tätä painiketta käytetään kelaamaan nauhaa vasemmanpuoleiselta kelalta oikeanpuoleiselle kelalle. Eteenpäinkelaustoiminto ei kuulu vapautusmekanismin piiriin, joten eteenpäinkelausta ei saa jättää käyntiin nauhan loputtua, koska muuten nauhan vetomoottori saattaa vioittua ylikuumetessaan.

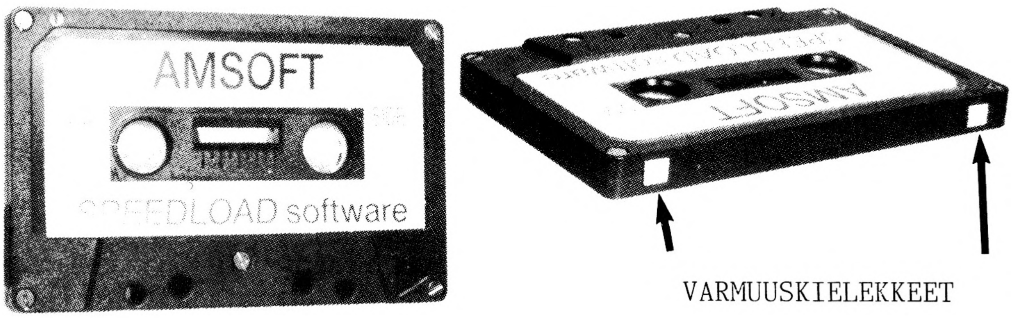
Pysäytys/kasetinpoistopainike [STOP/EJECT]

Tällä painikkeella voidaan pysäyttää mikä hyvänsä käynnissä olevista kasettitoiminnoista, jolloin kaikki kasettinauhurin käyttöpainikkeet palautuvat alkuasentoonsa. Jos vapautat painikkeen ja painat sitä uudelleen, avautuu kasettipesän kansi. Kasetti voidaan laittaa paikalleen kasettipesään tai ottaa pois kasettipesästä kannen ollessa auki. Kasettia ei voida ottaa pois kasettipesästä ennen kuin nauhan vetomekanismi on pysäytetty.

Taukopainike [PAUSE]

Tätä mekaanista taukopainiketta voidaan käyttää [PLAY]-, tai [REC]- ja [PLAY]-painikkeiden kanssa. Painiketta ei pidä käyttää silloin, kun tietokone on lukemassa tietoja nauhalta tai kirjoittamassa niitä nauhalle, koska se aiheuttaa virhetilanteen ja virheilmoituksen. Mekaanisen taukopainikkeen käyttö jää melko vähäiseksi, koska kaikki toisto- ja äänitystoimintojen aikana esiintyvät tauot hoidetaan CPC 464:n ohjelmiston sisäisillä käskyillä.

2.2 KASETIN TAKAREUNASSA OLEVAT VARMUUSKIELEKKEET



Kuva 2.3. Kasetin takareunassa olevat varmuuskielekkeet

Kaikissa tavallisissa C-kaseteissa on pienet varmuuskielekkeet niiden kapeassa takareunassa. Kumpaakin kasetin puolta varten on oma kielekkeensä. Jos esim. 1-puolta varten oleva kieleke poistetaan, ei kasettinauhurilla voi äänittää ohjelmia 1-puolelle. Tällä tavalla voit estää, etteivät ohjelmat ja tiedot pyyhkiydy vahingossa pois kasetilta. Muovikielekkeen voi irrottaa esim. pienellä ruuvimeisselillä. Kun kieleke poistetaan, ei [REC]-painike painu alas kasetin ollessa paikallaan kasettipesässä.

Kasetin takareunassa on siis yksi kieleke kumpaakin kasetin puolta varten. Jos haluat suojella kasetin molemmat puolet, on sinun irrotettava molemmat kielekkeet. Jos myöhemmin haluat äänittää ohjelmia sellaiselle kasetille josta muovikieleke (tai molemmat kielekkeet) on poistettu, voit peittää aukon pienellä teipinpalalla. Peitä aukko siten, että teippi ylettyy reilusti aukon yli. Tämän jälkeen kasetille voi äänittää ohjelmia aivan normaalisti.

2.3 KASETIN LAITTO KASETTINAUHURIIN

Kuvassa 2.1 on esitetty, kuinka CPC 464:n mukana seuraava esittelykasetti laitetaan oikealla tavalla kasettinauhuriin.

Nauha on kelattava ensin aivan alkuun (oikeanpuoleiselta kelalta vasemmanpuoleiselle kelalle). Ellei nauhaa ole kelattu, paina takaisinkelauspainiketta [REW], kunnes nauha pysähtyy kelauduttuaan kokonaan vasemmanpuoleiselle kelalle. Jos nauha on päässyt vahingossa löystymään siten, että kasetin etureunassa olevasta nauha-aukosta näkyy löysällä oleva nauhasilmukka, on nauha kelattava kasetille kiertämällä toista nauhakelaa esim. lyijykynällä ennen kuin kasetti laitetaan kasettipesään. Muussa tapauksessa nauha voi vioittua ja nauhalla olevat tiedot saattavat hävitä.

Tietokoneen kasetteja käsiteltäessä kannattaa muistaa, ettei kasetteja voi käyttää yhtä huolettomasti kuin tavallisissa äänentoistoon käytettävissä kasettinauhureissa, joilta musiikki kuuluu vielä senkin jälkeen, jos nauhaan tulee esimerkiksi ryppyjä. Tietokoneen kasettinauhurin kaseteissa asia on toisin: kasetteja on käsiteltävä huolellisesti ja varovasti, sillä kasetti tuskin toimii enää luotettavasti, jos nauhaan pääsee muodostumaan ryppy tai muu vastaava vika.

Jos vioitat nauhaa esimerkiksi jättämällä kasetin nauha-aukosta löysällä olevan nauhasilmukan kasettipesän kannen väliin mutta ohjelman lataaminen kasetilta tai ohjelman ajaminen kasetilta yhä edelleen onnistuu, sinun pitäisi välittömästi tallettaa ohjelma uudelleen SAVE-käskyllä käyttämättömälle ja vioittumattomalle nauhalle (niin kauan kun ohjelma on vielä tallessa CPC 464:n muistissa) ja heittää vioittunut nauha menemään ennen kuin kiusaus tulee liian suureksi käyttää kasettia uudelleen.

2.4 ESITTELYKASETTI

CPC 464:n mukana toimitetulla esittelykasetilla on useita esittelyohjelmia. Ne esittelevät CPC 464:n ääni- ja grafiikkamahdollisuuksia sekä koneen sisäistä ohjelmistoa eli AMSTRADin BASIC-kieltä ja koneen käyttöjärjestelmää.

Käyttöjärjestelmään kuuluu eräänä osana itse kasetin käyttöohjelmisto, johon kuuluu joukko lyhyitä komentoja. Niillä tiedot ja ohjelmat luetaan kasetilta muistiin ja tallennetaan muistista kasetille. LOAD- ja RUN-käskyt ovat useimmin käytettyjä käskyjä.

CPC 464:n toimintoihin kuuluu joukko erikoistoimintoja, joiden tarkoituksena on tehdä tietokoneen käyttö ja näppäimistöltä tapahtuva syöttö mahdollisimman yksinkertaiseksi, kunnes opit itse käyttämään näppäimistön eri mahdollisuuksia. Kytke virta tietokoneeseen, jolloin kuvaruudussa näkyy alkuviesti sekä tietokoneen valmiutta osoittava viesti **Ready** .

Laita kasetti paikalleen kuvan 2.1 osoittamalla tavalla, nollaa nauhalaskijan lukemaksi "0000" painamalla nauhalaskijan vieressä olevaa pientä nollauspainiketta. Paina seuraavaksi [CTRL]-näppäintä ja pidä se koko ajan alhaalla. Paina nyt samanaikaisesti numeronäppäimistön oikeassa alakulmassa olevaa pientä [ENTER]-näppäintä. Ruutuun ilmestyy seuraava viesti:

RUN"
Press PLAY then any key:

Tämä on yksi esimerkki ns. laajennusmerkistä, joka tuli ensimmäistä kertaa lyhyesti vastaan luvussa 1, jossa käsiteltiin KEY-käskyä. Tämä tarkoittaa sitä, että tietokone ohjelmoidaan suorittamaan kokonainen käskyjoukko, jonka se sitten suorittaa painettaessa esim. yhtä tai kahta näppäintä. Tällä tavalla voidaan nopeuttaa esim. ohjelmointityötä, koska usein esiintyvät käskyyhdistelmät saadaan käyttöön yhdellä näppäimen painalluksella.

Olisit myös voinut näppäillä käskyn RUN" ja painaa sen jälkeen [ENTER]-näppäintä, mutta on tietenkin helpompaa painaa vain kahta näppäintä saman lopputuloksen aikaansaamiseksi.

Useimmissa tapauksissa näppäimistön kaksi [ENTER]-näppäintä suorittavat saman tehtävän, mutta pienemmän [ENTER]-näppäimen toiminta voidaan muuttaa tietyissä olosuhteissa (eli näppäimen toiminta voidaan määritellä uudelleen), jolloin näppäin saadaan suorittamaan muunlaisia toimintoja ohjelmista saatavilla käskyillä. Tätä asiaa käsitellään tarkemmin myöhemmin käsikirjassa olevan luvun "Käyttäjän määriteltävissä olevat näppäimet" kohdalla.

Tietokone pyytää näytössä olevalla viestillä, että painaisit toistopainiketta [PLAY]. Tämä painike löytyy kasettinauhurin käyttöpainikkeiden joukosta. Paina nyt painiketta, kunnes se lukittuu kunnolla ala-asentoon. Viestin loppuosassa tietokone pyytää sinua "painamaan mitä hyvänsä näppäintä" (press any key). Tämä on hyvin yleinen tietokoneiden yhteydessä käytettävä sanonta, ja se saattaa johtaa harhaan, jos sitä tulkitsee liian hätäisesti. Sitä käytetään yksinkertaistamaan tiettyä käskyjonoa, jolla ohjelma saadaan etenemään seuraavaan toimintoon ilman, että kaikkia käskyjä tarvitsee erikseen määritellä.

Tarkkaan ottaen olisi oikeammin, jos viestissä sanottaisiin "paina mitä muuta näppäintä hyvänsä paitsi [SHIFT]-, [CAPS LOCK]-, [CTRL]- tai [ESC]-näppäintä tai jotakin kasetin käyttöpainiketta", mutta kaikkien tietokoneiden kanssa käytetään tiettyjä yksinkertaistuksia. Edellä oleva pätee kaikkialla tässä käsikirjassa tai AMSOFTin ja muiden toimittajien toimittamissa ohjelmissa sellaisessa kohdassa, jossa lukee "Press any key" (paina mitä hyvänsä näppäintä).

Näppäimen painallus ei aiheuta merkin tulostumista ruutuun vaan se käynnistää kasettinauhurin vetomoottorin. Ellei nauha lähde liikkeelle, paina jotakin muuta näppäintä (ota tavaksesi painaa suurta [ENTER]-näppäintä) ja tarkista, ettet ole vahingossa painanut kasettinauhurin taukopainiketta [PAUSE] alas.

Jos painat useampaa kuin yhtä näppäintä, ei tietokone ota huomioon muita näppäinten painalluksia sen jälkeen kun se on aloittanut ohjelman lataamisen kasetilta muistiin.

Huomaa, ettet ole määritellyt mitään erityistä ohjelman nimeä. Niin kauan kun käskyn RUN" perässä ei ole mitään muuta tekstiä samalla rivillä, etsii tietokone ensimmäisen nauhalta löytämänsä ohjelman ja alkaa ladata sitä kasetilta muistiin. Kun tietokone on löytänyt ensimmäisen oikein äänitetyn ohjelman nauhalta, ilmestyy kuvaruutuun seuraava viesti:

Loading WELCOME 1 block 1

Tällä viestillä tietokone ilmoittaa sinulle, että se on löytänyt ensimmäisen lohkon (block) ohjelmasta, jonka nimi on "WELCOME 1". Jokainen ohjelma on tallennettu nauhalle perättäisinä tietolohkoina (yhden lohkon enimmäispituus on 2 kilotavua), jotka tietokone lukee nauhalta muistiin. Tietokone tunnistaa kunkin tietolohkon erikseen nauhalta, ja näet näytössä olevasta viestistä, mitä lohkoa tietokone milloinkin lukee muistiinsa. Nauha pysähtyy hetkellisesti ja käynnistyy uudelleen jokaisen tietolohkon jälkeen, ja tietokone "päivittää" näytössä olevan viestin heti nauhan käynnistyttyä uudelleen, jolloin viestissä näkyy tietokoneen parhaillaan lukeman uuden lohkon numero.

Jos lukuvaiheessa esiintyy virhe esim. siksi, etteivät tiedot ole tallentuneet nauhalle kunnolla, ilmestyy näyttöön virheilmoitus (joista on luettelo liitteessä VIII). Virheilmoituksessa kerrotaan, mistä vaikeudet johtuvat. Yleensä ei ole mahdollista tehdä mitään muuta kuin yrittää käynnistää ohjelman lataaminen uudelleen, kunnes ohjelma latautuu virheittä.

Kun ohjelma on alkanut latautua esittelykasetilta tietokoneen muistiin, noudata näytössä näkyviä ohjeita - ja esittelykasetti hoitaa loput.

2.5 KAKSI TIEDONSIIRTONOPEUTTA TALLETUKSEEN JA LUKUUN

CPC 464:ssä on käytettävissä kaksi nopeutta, joilla tiedot voidaan tallettaa muistista kasetille tai lukea kasetilta muistiin. Nopeudet ovat 1000 baudia (bittiä sekunnissa) ja 2000 baudia. Suurempi nopeus on siis kaksinkertainen pienempään nopeuteen verrattuna, vaikkakin varmuusmarginaalista joudutaan hieman tinkimään, koska tiedot eivät tallennu aivan yhtä varmasti halvoille kaseteille niiden nauhoissa esiintyvien epätasaisuuksien vuoksi. Suurempaa nopeutta käytettäessä saattaa myös esiintyä virheitä siirrettäessä tietoja kumpaan hyvänsä suuntaan, koska äänipäiden säätö voi olla hieman erilainen nauhurista riippuen.

Kun ohjelmat talletetaan muistista kasetille ja luetaan takaisin kasetilta muistiin samalla nauhurilla, pitäisi myös suuremman nopeuden osoittautua käytännössä luotettavaksi, kunhan vain nauhurissa käytetään riittävän laadukkaita kasetteja. Ohjelmien lukeminen useimmilta myytävänä olevilta ohjelmakaseteilta pitäisi onnistua virheittä myös suuremmalla nopeudella. Kaikki tällaiset ohjelmat kannattaa kuitenkin äänittää molemmilla nopeuksilla. Tietokoneen lukunopeus asettuu automaattisesti samaksi kuin millä tiedot on talletettu nauhalle. Kun ohjelmaa talletetaan nauhalle SAVE-käskyllä, on tietokoneelle ilmoitettava, jos talletuksessa halutaan käyttää suurempaa tiedonsiirtonopeutta. Muussa tapauksessa tietokone valitsee oletusarvona pienemmän nopeuden.

Jos haluat käyttää suurempaa tiedonsiirtonopeutta tietojen ja ohjelmien tallettamiseen on sinun ensin varmistettava, että tietokone on valmis ottamaan vastaan käskyjä (kuvaruudussa näkyy **Ready** -ilmoitus). Tämän jälkeen näppäillään:

```
SPEED WRITE 1
```

Pienempään nopeuteen palataan joko nollaamalla tietokone (jolloin myös kaikki tiedot häviävät muistista) tai näppäilemällä (kun kuvaruudussa näkyy **Ready** -ilmoitus):

```
SPEED WRITE 0
```

2.6 OHJELMIEN JA TIETOJEN TALLETTAMINEN KASETILLE

BASIC-kielessä on useampia komentoja, jotka liittyvät tietojen tallettamiseen kasetille. Käsitelkäämme niitä nyt lyhyesti muutamien esimerkkien valossa.

2.6.1 SAVE "<tiedoston nimi>"

Tietojen tallettaminen kasetille käy kaikkein mutkattomimmin SAVE-käskyllä, kun CPC 464:n näytössä näkyy **Ready** -ilmoitus ohjelman suorittamisen tai listauksen jälkeen. Käytämme nyt SAVE-käskyn kanssa esimerkkiohjelmana sitä lyhyttä ohjelmaa, joka tulostaa ruutuun tietokoneen merkkivalikoiman (ohjelmaa käsiteltiin luvussa 1).

Lainausmerkkeihin kirjoitettava tiedoston nimi voi olla mikä hyvänsä yhdistelmä kuudestatoista näppäimistön merkistä (mukaan lukien välilyönnit). Jos yrität antaa pitemmän nimen, tietokone hylkää seitsemännentoista ja sen jälkeen olevat merkit. Kun esimerkkiohjelma on tietokoneen muistissa, näppäile:

SAVE "MERKIT"

Tietokone vastaa antamaasi käskyyn pyytämällä sinua painamaan kasettinauhurin äänitys- ja toistopainiketta ja sen jälkeen mitä hyvänsä näppäintä tulostamalla näyttöön ilmoituksen:

Press REC and PLAY then any key:

Muistathan, mitä sanonta "mitä hyvänsä näppäintä" tarkoittaa. Asiaa käsiteltiin muutama sivu aikaisemmin. Nauha käynnistyy ja tietokone alkaa tallettaa ohjelmaa nimellä MERKIT, jos olet toiminut tietokoneen pyytämällä tavalla.

TÄRKEÄÄ!

Paina mieleesi, ettei tietokone voi tietää, oletko todella painanut oikeita kasettinauhurin käyttöpainikkeita. Nauha kylläkin käynnistyy ja näyttää siltä kuin tietokone tallettaisi ohjelman kasetille, jos painat vain toistopainiketta [PLAY]. Ohjelma tallettuu kasetille vain silloin, kun painat molempia painikkeita yhtäaikaa.

VAROITUS!

Jos painat vahingossa [REC]- ja [PLAY]-painikkeita halutessasi lukea eli ladata ohjelman kasetilta tietokoneen muistiin, pyyhkiytyvät kaikki kasetilla olevat ohjelmat lopullisesti pois nauhalta. Ellei pyyhkiytymistä keskeytetä [ESC]-näppäintä painamalla, nauha kelautuu kasetin loppuun ja nauha pyyhkiytyy aivan tyhjäksi, koska tietokoneelta ei tule nauhan pysäytyskäskyä. Normaalisti pysäytyskäsky saadaan, kun tietokone löytää nauhalta etsimänsä ohjelman. Jos pelkäät kadottavasi tietoja tai ohjelmia edellä mainitusta syystä, voit välttää sen poistamalla kasetin kapeasta takareunasta varmuuskielekkeen (tai molemmat kielekkeet), jolloin kasettinauhurin äänityspainike ei painu alas. Tämä on parempi tapa kuin opetella asia "kantapään kautta" menettämällä ehkä arvoikkaitakin ohjelmia.

CPC 464:ssä on käytettävissä neljä eri tapaa, joilla tiedostoja (eli ohjelmia tai tietoja) voidaan tallettaa kasetille SAVE-käskyllä. Olemme juuri käyneet läpi kaikkein yleisimmän tavan, mutta jäljellä on vielä kolme vaihtoehtoa hieman erikoisempiin käyttötilanteisiin.

2.6.2 SAVE "<tiedoston nimi>",A

Menetelmä on sama kuin kohdassa 2.6.1 käytetty sillä erotuksella, että käskyrivin lopussa oleva liite **,A** ilmoittaa tietokoneelle, että ohjelma tai tiedot on talletettava ASCII-tekstitiedoston muodossa.

Tätä menetelmää käytetään talletettaessa tekstinkäsittely- tai muilla sovellutusohjelmilla luotuja tiedostoja. Tätä asiaa käsitellään tarkemmin siinä vaiheessa kun tulemme kyseisten sovellutusten kohdalle.

2.6.3 SAVE "<tiedoston nimi>",P

Käskyrivin lopussa oleva liite **,P** ilmoittaa tietokoneelle, että tiedot on talletettava kasetille suojatussa (koodatussa) muodossa, jolloin kuka tahansa ei voi helposti lukea ohjelmaa kasetilta RUN-käskyllä ja pysäyttää sen suoritusta [ESC]-näppäintä käyttämällä. Tällä tavalla talletetut ohjelmat saadaan uudelleen käyttöön kasetilta vain RUN- tai CHAIN-käskyillä. Jos sinusta tuntuu siltä, että aiot myöhemmin korjailla tai muuttaa ohjelmaa, sinun kannattaa tallettaa ohjelmasta yksi kopio myös suojaamattomassa muodossa.

2.6.4 SAVE "<tiedoston nimi>",B,<alkuosoite>,<pituus>[,<aloituskohta>]

Kun näppäilet SAVE-käskyn tässä muodossa, voit tallettaa kokonaisen tietolohkon binäärimuodossa nauhalle juuri siinä muodossa kuin se on tallennettu tietokoneen RAM-muistiin. Tietokoneelle on kerrottava, mistä kohdasta tallennuksen kohteena oleva muistilohko alkaa ja kuinka pitkä se on. Samoin käskyn yhteydessä on määriteltävä muistiosoite, josta tiedoston suorittaminen aloitetaan, jos se ajetaan ohjelmana RUN-käskyllä.

Tätä binäärimuotoista talletustapaa voidaan käyttää silloin, kun näytöstä halutaan tallettaa tietoja kasetille suoraan "näyttövedoksen" muodossa. Eräs sen pääkäyttöalueista on "otsikkojaksojen" muodostaminen pitkissä ohjelmissa, joilla saadaan vaihtelua pitkien ohjelmien yksitoikkoiseen latausvaiheeseen.

2.6.5 Nimettömät tiedostot ja CAT-käsky

Jos talletat jonkun tiedoston SAVE-käskyllä antamatta sille nimeä näppäilemällä:

```
SAVE""
```

niin BASIC tallettaa ohjelman nimettömänä tiedostona (Unnamed file). Kasetille voidaan tallettaa peräkkäin niin monta samannimistä tiedostoa (mukaan lukien nimettömät tiedostot) kuin kasetille vain mahtuu. Tässä suhteessa kasetti eroaa tietolevystä, koska tietolevylle on aina annettava uusi aikaisemmin käyttämätön nimi tiedostoa talletettaessa.

Sinun kannattaa antaa ohjelmillesi sellaiset nimet, että näet niistä välittömästi mitä ohjelmat pitävät sisällään. Muussa tapauksessa halutun ohjelman löytäminen kasetilta on varsin hankalaa. Nimeen kannattaa myös lisätä jonkinlainen päivämäärämerkintä, jolloin näet heti, mikä ohjelmista on tuorein ja parhaiten ajantasalla oleva.

Kasetin sisällysluettelo

Voit pyytää sisällysluettelon kasetilla olevista ohjelmista antamalla CAT-käskyn ja noudattamalla ruutuun ilmestyviä ohjeita:

Press PLAY then any key:

Kun siis painat PLAY-painiketta ja sen jälkeen mitä hyvänsä näppäintä, saat kuvaruutuun sisällysluettelon kasetilla olevista ohjelmista. Tietokone tulostaa tiedostojen nimet isoilla kirjaimilla, ja nimen perässä on tietolohkojen lukumäärä. Lisäksi ruutuun tulostuu kullekin riville yksi merkki joka ilmoittaa, minkälainen tiedosto on kysymyksessä:

$ on vakiomuotoinen BASIC-ohjelma
% on suojattu BASIC-ohjelma
* on ASCII-muotoinen tekstitiedosto
& on binäärimuotoinen tiedosto

Rivin lopussa oleva **Ok** on merkki siitä, että tiedosto on luettavissa ja että se olisi latautunut kasetilta tietokoneen muistiin, mikäli tietokonetta olisi pyydetty lataamaan se muistiin. CAT-käskyn antaminen ei vaikuta sillä hetkellä tietokoneen muistissa olevaan ohjelmaan.

2.7 LUKUVIRHEET

Jos näyttöön ilmestyy lukuvirheestä ilmoittava viesti **Read error** on se merkki siitä, että CPC 464:n yrittäessä ladata ohjelmaa tai tietoja kasetilta muistiin on esiintynyt lukuvirhe. Tällöin nauha ei pysähdy vaan kasettinauhuri jatkaa toistotilassa ([PLAY]-painike alhaalla) ja tietokone lukee edelleen lukuvirheen jälkeen nauhalta löytämiään lohkoja (block). Erona normaaliin tilanteeseen on se, ettei tietokone yritä ladata (LOAD) lohkoja muistiin ellei niitä ole määritelty sen ohjelman lohkoksi 1, jota tietokone alunperin yritti ladata (epäonnistuen) kasetilta muistiin.

Tämä merkitsee sitä, että voit pysäyttää nauhan lukuvirheen tapahduttua kasettinauhurin [STOP/EJECT]-painikkeella ja kelata nauhan alkuun [REW]-painikkeella, ja painaa sen jälkeen [PLAY]-painiketta uudelleen. Nyt tietokone yrittää uudelleen ladata ohjelman, jota luettaessa lukuvirhe esiintyi. Uudella yrityskerralla lukeminen saattaa sujua jo paremmalla onnella.

Lukuvirheet voivat aiheutua useammastakin syystä. Kaikkein yleisin syy on kuitenkin se, että kasetin nauha on vahingossa vioittunut, eli nauhan herkkä äänityspinta on päässyt rypistymään, venymään tai muulla tavalla muuttamaan muotoaan. Nauha on saattanut vioittua myös siksi, että tietokoneesta on katkaistu virta kasettinauhurin [PLAY]-painikkeen tai [REC]- ja [PLAY]-painikkeiden ollessa painettuna alas. Tämä saattaa aiheuttaa lukuvirheen siksi, että kun painike on painettu alas, pitää nauhamekanismi nauhaa äänipäätä vasten. Kun virta nyt katkaistaan tietokoneesta, voi äänitys/toistopään kautta kulkea lyhyt sähköpulssi virtalähteen tyhjentyessä. Vaikka itse nauha pysyykin muuttumattomana, saattaa tämä virrankatkaisun yhteydessä esiintyvä pulssi (ja samanlainen pulssi joka esiintyy silloin kun virta kytketään tietokoneeseen) sekoittaa tehokkaasti kyseisessä nauhan kohdassa olevat tiedot, jolloin tietojen lukeminen käy mahdottomaksi. Lisäksi nauha voi rypistyä, jos se joutuu olemaan pitkän aikaa puristuksissa vetoakselin ja puristusrullan välissä.

Lukuvirheitä voi esiintyä myös silloin, jos taukopainiketta [PAUSE] painetaan kasettinauhurin ollessa äänitys- tai toistotilassa, tai jos nauha on äänitetty alunperin jollakin toisella CPC 464:llä, jonka kasettinauhurin äänipäiden säätö ei ole aivan tarkasti kohdallaan.

Lukuvirheitä voi esiintyä joskus myös aivan sattumanvaraisista syistä. C-kasettia ei ole alunperin suunniteltu tietojen tallennusvälineeksi, joten sen rakenteessa ei ole huomioitu kaikkia niitä vaatimuksia, jotka kalliimmat "ammattimaiset" nauhatallennusjärjestelmät täyttävät.

Siitä huolimatta kasetti on osoittautunut erinomaiseksi jokapaikan tallennusvälineeksi huokeana tiedontallennusvälineenä. Nauhalla olevien magneettisten hiukkasten fyysiset ominaisuudet (lähinnä koko) ja nopeus, jolla nauha kulkee äänipään ohi, asettavat tietyt rajoitukset sille nopeudelle, jolla tietoja voidaan siirtää nauhan ja tietokoneen välillä. Jos nopeuksia yritetään nostaa liian suuriksi varmuusrajan yläpuolelle, on siitä seurauksena toiminnan luotettavuuden heikkeneminen. Tämä pätee erityisesti halvoille ohjelmakaseteille, joiden kopioinnissa käytetään massavalmistusmenetelmiä.

HUOM!

Muiden kuin CPC 464-tietokoneiden ohjelmia sisältäviä kasetteja ei voida käyttää CPC 464:ssä, koska ohjelmat eivät lataudu kasetilta tietokoneen muistiin. Kasetit voivat kylläkin näyttää samanlaisilta, ja niistä saattaa jopa kuulua samanlaisia "ääniä" toistettaessa niitä äänentoistoon tarkoitetuilla kasettinauhureilla, mutta ohjelmat eivät lataudu kasetilta muistiin eikä ohjelmia voida ajaa. Jos löydät tällaisen "vieraan" koneen kasetin jonka ohjelma latautuu ja jonka voit ajaa CPC 464:llä, olemme varsin kiinnostuneita kuulemaan yksityiskohtaiset tiedot tietokoneen tyypistä ja kyseisestä ohjelmasta.

2.8 KASETIT

CPC 464:n kasettinauhurissa tulisi käyttää vain C12-kasetteja (6 minuuttia kummallakin puolella) tai korkeintaan C30-kasettia, vaikka nauhuri toimii mukisematta kaikenlaisilla kaseteilla, jotka voivat olla nauhapituudeltaan jopa C90-tyyppisiä (45 minuuttia kummallakin puolella). Pitkän nauhan loppuun talletettuja ohjelmia on vaikea löytää, ellet halua nähdä vaivaa niiden löytämiseksi, koska ohjelman löytyminen saattaa joskus kestää pitkänkin ajan. Voit tietenkin merkitä paperille kasetilla olevien ohjelmien nimet ja vastaavat nauhalaskijan lukemat ohjelmien alku- ja loppukohtien kohdalta, mutta vaikeuksia saattaa ilmetä siinä vaiheessa kun haluat äänittää uuden ohjelman jonkin vanhan ohjelman päälle. Tällöin vanhan ohjelman alkukohta on paikannettava tarkasti ja pidettävä visusti huoli siitä, ettei uusi ohjelma äänity myöskään loppupäästään minkään sellaisen ohjelman päälle, jonka haluat edelleen säilyvän nauhalla.

Pääset kaikkein vähimmällä vaivalla, kun talletat kullekin kasetille mahdollisimman vähän ohjelmia. C12-kasetit ovat myös suhteellisen halpoja, eikä C12-kasetin heittäminen menemään sen mahdollisesti vioittuessa ole niin vaikeaa kuin esim. C90-kasetin, jossa on edelleen jäljellä runsaasti vioittumatonta nauhaa.

Lopuksi sinun on vielä syytä muistaa, että lähes kaikki myytävät ohjelmat on tarkoin suojattu tekijänoikeuslain säännösten perusteella. Kaseteilla toimitettavia ohjelmia ei saa kopioida, ellei siitä ole erikseen sovittu ohjelman myynnin yhteydessä (joistakin ohjelmista jopa kehoitetaan ottamaan varmuuskopio itselleen). Ohjelman kopiointi on myös kiellettyä, vaikka se tulisi "vain kaverille". Tekijänoikeuslain säännöksiä ollaan parhaillaan tarkistamassa koskemaan kaikenlaista luvatonta ohjelmien kopiointia. Vaikka oikeustapauksia onkin tähän mennessä ollut vain muutamia, tulee tilanne muuttumaan merkittävästi lähivuosien aikana, ja määräykset saattavat astua voimaan jopa takautuvasti.

LUKU 3: BASICIN ALKEET

Tässä luvussa esitellään lyhyesti AMSTRADin BASIC-kielellä kirjoitettuja ohjelmia.

Luvussa käsiteltäviä asioita ovat:

- Muotosäännöt ja muotokuvaukset
- PRINT-käskyt ja eri tulostustavat
- Vyöhykkeet (ZONE)

3.1 BASIC-KIELEN PERUSASIOITA

CPC 464:n BASIC-kielen ja tietokoneen sisäisen toiminnan välisen riippuvuuden perusasiat on esitetty liitteessä II. Jos et ole aikaisemmin ohjelmoinut tietokonetta, yritämme auttaa sinua alkuun. Joudumme kuitenkin tekemään eräitä olettamuksia, jotka saattavat aiheuttaa hämmennystä aivan aloittelijoille. Jos näin käy, sinun kannattaa tutustua kaupoista saataviin ohjelmointikirjoihin. Niissä on hyviä neuvoja aloittelijoille tietokoneiden ohjelmointiin liittyvistä perusasioista.

Sinun pitäisi pystyä kahlaamaan tämä luku läpi seuraamalla lukuun liittyviä yksinkertaisia harjoituksia, eikä sinun tarvitse ymmärtää aivan kaikkea mitä ohjelmissa tapahtuu. On tietenkin niin, että mitä enemmän opit, sitä helpompaa on jatkossa.

CPC 464:n "sisällä" on BASIC-kieli, joka on käyttövalmiina heti kun kytket virran tietokoneeseen. Tiedät tämän siitä, että näyttöön ilmestyy viesti **Ready** .

BASIC on kaikista kielistä helpoin oppia. Siinä käytetään tarkasti määriteltyjä sanoja ja "kielioppia" ja se toimii täysin johdonmukaisesti - jos vain ymmärtää sen säännöt.

AMSTRADin BASIC pystyy suorittamaan luvussa 8 luetellut käskyt. Jokainen käsky määritellään yhdellä tai useammalla avainsanalla, johon voi liitttyä useitakin parametrejä. Aina ei kuitenkaan tarvitse käyttää kaikkia käytettävissä olevia parametrejä. Yleisesti ottaen voidaan sanoa, että kukin parametri voi olla jonkinlainen lauseke, johon kuuluu vakioita, muuttujia tai toimintoja. Numeroiden ja kirjainten yhdistelmiä kutsutaan merkkijonoiksi, ja niissä voi olla useamman tyyppisiä numerotietoja kuten desimaali-, heksadesimaali- ja binäärimuotoisia numerotietoja.

Kasetilla olevia tiedostoja (esim. ohjelmia) käsitellään peräkkäisjärjestyksessä eli toinen toisensa perään. Tämä eroaa esim. tietolevyjä käyttävästä järjestelmästä, jossa tiedosto voidaan valita suoraan muiden joukosta tarvitsematta ensin kulkea tarpeettomien tiedostojen ohi.

3.2 BASIC-OHJELMAN RAKENNE

BASIC-ohjelmassa käskyt ovat riveillä. Yksi rivi voi koostua useammasta kaksoispisteellä erotetusta käskystä. Yhden käskyrivin maksimipituus on 255 merkkiä. Merkki voi olla numero, kirjain tai välilyönti. Kun käskyrivit näppäillään suoraan näppäimistöltä, niiden alkuun ei laiteta rivinumeroa. Yleensä käskyrivien alkuun kuitenkin laitetaan rivinumero, jolloin numeroidut käskyrivit tallentuvat tietokoneen ohjelmamuistiin, josta tietokone sitten suorittaa ohjelman käskyn saatuaan rivinumeroiden osoittamassa järjestyksessä.

Muistissa olevaan ohjelmaan voidaan lisätä tai siitä voidaan poistaa rivejä. Ohjelmarivejä voidaan niin ikään korjata ja muotoilla uuteen uskoon. CPC 464:n BASIC-järjestää automaattisesti ohjelmarivit numerojärjestykseen alkaen pienimmästä rivinumerosta, ennen kuin ohjelma ajetaan RUN-käskyllä tai ruutuun pyydetään listaus LIST-käskyllä. Tämä ei riipu lainkaan siitä, missä järjestyksessä rivinumerot on näppäilty koneeseen.

3.3 OHJELMARIVIEN NÄPPÄILY

BASIC hyväksyy ohjelmaan rivejä, joilla on enintään 255 merkkiä. Jokainen rivi on päätettävä painamalla [ENTER]-näppäintä. Kun riviä näppäillään, voidaan kyseistä riviä korjailla tavalliseen tapaan sekä käyttämällä myös kopiointikohdistinta, jolloin riville voidaan lisätä merkkejä jostakin muusta näytössä näkyvästä kohdasta; katso luvun 1 kappaletta 1.2.7.

Kaikki avainsanat täytyy erottaa toisistaan jollakin erottimella. Se voi olla välilyönti, laskutoimituksen merkki (+ - jne.) tai jokin muu hyväksytty merkki. Näin on tehtävä siksi, että AMSTRADin BASIC sallii sellaisten muuttujien käytön, joissa avainsana (tai varattu sana) muodostaa osan muuttujasta. Varattu sana ei tietenkään voi olla muuttuja, ellei sen ympärille laiteta joitakin erotusmerkkejä, joilla tietokonetta estetään tunnistamasta käskyn alku ja loppu.

Avainsanat voidaan näppäillä joko pienillä kirjaimilla tai suurilla kirjaimilla ([SHIFT]-näppäintä käyttämällä).

PRINT-käskyn sijasta voidaan käyttää sen lyhennettyä muotoa eli kysymysmerkkiä **?** , jolloin sen kanssa ei tarvita erotinta. Myös matemaattisten laskutoimitusten merkit (+ - * / MOD) toimivat avainsanan erottimena, joten AMSTRADin BASIC hyväksyy alla olevan käskyrivin vaikka sen käyttöä ei suositellakaan, koska se saattaa opettaa huonoille tavoille myös sellaisten ohjelmien näppäilyissä, joissa tarvitaan välilyöntejä:

```
for n= 1 to 50:?n:next
```

REM-käskyä käytetään haluttaessa kirjoittaa ohjelmaan huomautuksia. Myös REM-käsky voidaan lyhentää, ja sen tilalla voidaan käyttää heittomerkkiä ' ([SHIFT] 7).

Ohjelma ei ota huomioon ylimääräisiä välilyöntejä, joita voidaan käyttää ohjelmalistauksen muotoiluun haluttaessa osoittaa korostetusti, missä esim. eri silmukat sijaitsevat.

3.4 TERMINOLOGIA

BASIC-kielessä on käytettävä muodollista mutta tarpeeksi yksinkertaista terminologiaa (eli sanontoja ja sanontatapoja), jotta eri käskyt ja avainsanat voitaisiin määritellä riittävän selkeästi. Jokainen käsky on esitetty tässä käsikirjassa siinä muodossa jossa se näkyy kuvaruudussa näppäiltäessä käsky näppäimistöltä. Lisäksi kunkin käskyn yhteydessä on esitetty siihen mahdollisesti liittyvät muuttujat ja valinnaiset (= ei pakolliset) osat, jotka on esitetty erilaisissa sulkumerkeissä tai lainausmerkeissä. Näille käskyn "lisäosille" on annettu erilaisia nimiä, ja ne on merkitty aina < >-merkkien sisään. Esimerkki: käskyn yhteydessä tarvitaan usein jotakin numeroa tai lukua, jolloin tämä on esitetty muodossa:

<numerolauseke>

Kaikki sellainen, mitä ei ole kirjoitettu < >-merkkien sisään, on näppäiltävä täsmälleen käsikirjassa esitetyssä muodossa. Esimerkiksi STOP-käsky on näppäiltävä muodossa:

STOP

Jos käskyn määrittelyosaan kuuluu valinnainen osa, on tämä valinnainen osa laitettu hakasulkeisiin []. Esimerkki: jos numerolauseke on valinnainen, on sen esitystapa seuraava:

[<numerolauseke>]

Jos käskyyn liittyvä valinnainen osa voi toistua (se voi toistua useampaan kertaan tai ei yhtään kertaa), on jälkimmäisen hakasulun jälkeen lisätty tähtimerkki * . Esimerkki: numerojono, johon tarvitaan vähintään yksi numero, esitettäisiin muodossa:

<numero>[<numero>]*

Tällainen lauseke saattaisi olla esimerkiksi:

34
tai 344
tai 345678 jne.

Monasti käskyjen yhteydessä käytetään pilkulla erotettuja käskyn osia. Siitäkin käytetään lyhennettyä muotoa, jota kuvannee parhaiten seuraava esimerkki:

<lauseke>-jono tarkoittaa: <lauseke>[,<lauseke>]* tai:
[#]<luku>-jono tarkoittaa: [#]<luku>[,[#]<luku>]*

Tällainen voisi olla esimerkiksi:

3,4
tai 3,4,4
tai 3,4,5,6,7,8 jne.

"Jonossa" ei tarvitse välttämättä olla kuin yksi osa. Jos jonoon kuitenkin kuuluu useampia osia, on jokainen osa erotettava pilkulla, jolloin tietokone osaa käsitellä kunkin osan erikseen.

Numeroiden ja lukujen esitystapa

Numeroiden ja lukujen esittämiseksi käytetään useampia tapoja:

a. <skaalaamattomat luvut>

Nämä ovat lukuja, joissa ei ole eksponenttiosaa.

b. <skaalatut luvut>

Nämä ovat lukuja, jotka "korotetaan potenssiin" tai "skaalataan" käyttäen muotoa:

2E4 (kaksi kertaa kymmenen potenssiin neljä eli 2×10^4)

Eksponenttiosa voi olla positiivinen tai negatiivinen.

c. <tiettyyn kantalukuun perustuvat luvut>

Nämä ovat lukuja, joiden sanotaan olevan joko binäärilukuja tai heksadesimaalilukuja (katso liitettä II):

Desimaalimuoto (oletustila)	100	
Heksadesimaalimuoto	&64 tai &H64	("H" on valinnainen)
Binäärimuoto	&X1100100	("X" on pakollinen)

3.5 Harjoitus tekee mestarin - tutustukaamme PRINT-käskyyn

Tutkikaamme nyt seuraavien BASIC-kielisten käytännön esimerkkien avulla, kuinka edellä mainitut seikat toimivat käytännössä.

Useimmat seikat selviävät käyttämällä yksinkertaista BASIC-käskyä PRINT. Käskyksi sanotaan sellaista BASIC-avainsanaa tai lauseketta, joka saa tietokoneessa saman toiminnon aikaan siitä riippumatta, näppäilläänkö käsky suoraan näppäimistöltä vai sisältyykö se muistissa olevaan ohjelmaan. Käskyjä tarvitaan, jotta tietokonetta voitaisiin "käskeä" suorittamaan jokin toiminto, esimerkiksi:

PRINT FRE("") (ilmoittaa vapaan muistitilan)

Sinun on kerrottava CPC 464:lle kolme asiaa, jotta saisit sen vastaamaan kysymykseesi. Ja nämä asiat ovat:

1. Mihin haluat vastauksen ilmestyvän eli haluatko vastauksen kuvaruutuun, kirjoittimelle tai "muualle".
2. Sinun on annettava tietokoneelle tiedot, joiden kanssa sen on työskenneltävä.
3. Sinun on kerrottava tietokoneelle, mitä sen on tehtävä tiedoille.

PRINT-käskyllä tietokone saadaan tulostamaan käskyn suorittamisen seurauksena saatavat tulokset tietylle "tulostusreitille" (tai "tulostuskanavalle"). Tulostusreitti määritellään numerolla, joka voi olla väliltä 0 - 9. Siitä käytetään nimitystä <reittilauseke> BASIC-kielen rakennetta kuvaavissa selityksissä. Lausekkeessa annettava numero määrittelee siis kulloinkin käytettävän tulostusreitin.

Numerot 0 - 7 ovat tekstin tulostusreittejä, joilla tekstiä saadaan tulostumaan "ikkunoin". Ikkunat täytyy kuitenkin tätä ennen määritellä WINDOW-käskyllä.

Numero 8 on varattu kirjoittimen rinnakkaisliitäntäportille ja sitä voidaan käyttää vain silloin, jos CPC 464:ään on liitetty Centronics-yhteensopiva kirjoitin.

Numero 9 tarkoittaa kasetin tulostustiedostoa, joka on täytynyt jo aikaisemmin avata oikealla tavalla ohjelmassa.

Täten PRINT-käskyn lyhyt muoto (silloin kun ei käytetä muotoa PRINT USING) on:

```
PRINT [#<reittilauseke>],["tulostettava joukko"]
```

Hakasulkeet merkitsevät sitä, ettei sinun tarvitse välttämättä määritellä käskyn osaa <reittilauseke>, eikä sinun myöskään tarvitse määritellä PRINT-käskyssä tulostettavaa joukkoa. Jos siis jätät määrittelemättä nämä kohdat ja annat pelkän PRINT-käskyn, tulostaa tietokone pelkän tyhjän rivin (kokeile sitä). Ellet määrittele käskyssä mitään tiettyä tulostusreittiä, CPC 464 olettaa, että tarkoitat tulostusreittiä #0 eli kohdistimen alla olevaa kuvaruudun kohtaa. Kokeile seuraavaa riviä (ja muista painaa [ENTER]-näppäintä lopuksi, jotta tietokone ryhtyy käsittelemään antamiasi käskyjä):

```
PRINT "TERVE"
```

Tietokone vastaa antamaasi käskyyn tulostamalla ruutuun:

```
TERVE
```

Lainausmerkkejä "" ei käytetä tulostusreitin määrittelyyn. Lainausmerkkien käyttö on BASIC-kielessä tarkasti rajattu tulostettavan joukon alku- ja loppukohtien määrittelyyn. Näppäile seuraavaksi:

```
PRINT #0,"TERVE"
```

Tulos on nyt sama kuin edellisellä kerralla. Mutta jos näppäilet:

```
PRINT #4,"TERVE"
```

niin tietokone on sijoittanut tuloksen kuvaruudun vasempaan yläkulmaan, koska tämä kohta on kuvaruudun tulostusreitin nro 4 ensimmäinen sisääntulokohta, joka oletusarvona peittää koko tekstialueen, ellei sitä ole aikaisemmin määritelty WINDOW-käskyllä. Kaikki tekstit alkavat kuvaruudun vasemmasta yläkulmasta, kun tekstiä tulostetaan kuvaruudun tulostusreiteille. TERVE-sana tulostui juuri vasempaan yläkulmaan, koska tulostusreittiä 4 ei oltu vielä käytetty. Alkuviesti (eli nollausviesti) käyttää tulostusreittiä 0 (oletusarvoa), joten teksti tulostui ensimmäisellä kerralla näytössä näkyvien merkkien jälkeen ja toisella kerralla tulostusreitin 4 ensimmäiseen vapaaseen merkkipaikkaan.

Tämä AMSTRADin BASIC-kielen ominaisuus on erityisen kätevä, koska sen avulla voidaan muodostaa varsin monimutkaisia kuvia näyttöön yksinkertaisilla käskyillä ja määritteillä.

BASIC tulostaa kaiken sen, mitä laitat lainausmerkkien sisään. Lainausmerkkien sisään voidaan laittaa myös käskysanoja ja varattuja sanoja niiden vaikuttamatta tietokoneen toimintoihin. Jos näppäilet esimerkiksi:

```
PRINT "4*4"
```

niin tietokone tulostaa ruutuun:

```
4*4
```

Jos haluat, että BASIC suorittaa kertolaskun (4 kertaa 4), on numerot ja kertolaskun merkki * näppäiltävä sellaisessa muodossa, että BASIC pääsee suorittamaan laskutehtävän. Samalla tulos voidaan ohjata halutulle tulostusreitille. Ellei reittiä määritellä, käytetään oletusarvona jälleen reittiä numero 0. Näppäile siis:

```
PRINT 4*4
```

Tietokone suorittaa nyt laskutehtävän ja tulostaa ruutuun vastauksen:

```
16
```

Huomaa, että vastaus tulostuu yhden merkkipaikan päähän vasemmasta reunasta, koska BASIC varaa tämän tilan miinusmerkille (-), jolla negatiivinen tulos erotetaan positiivisesta tuloksesta.

PRINT-käskyllä on monia muitakin muotoja, joiden avulla käskyn suorittamisen seurauksena saatava tulos saadaan tulostetuksi ruutuun halutun mallin mukaan.

PRINT-käskyn yhteydessä oleva "tulostettava joukko" tarkoittaa sitä merkkijoukkoa, joka PRINT-käskyllä halutaan tulostaa. Se voi olla jokin numero, muuttuja tai merkkijonolauseke, mikä tarkoittaa aiemmin määriteltyä merkkijonomuuttujaa (esim. TERVE$) tai mitä hyvänsä muuta, joka on laitettu lainausmerkkien sisään.

PRINT USING-muotoa käytetään haluttaessa muotoilla ruutuun tulostuvat luvut tietyssä muodossa, jolloin numerosarakkeet ovat aina samassa kohdassa ja esim. desimaalipisteen jälkeiset desimaalinumerot voidaan jättää tulostamatta.

3.6 PRINT USING-MUOTO JA VYÖHYKKEET (ZONE)

Kun virta kytketään tietokoneeseen, BASIC asettaa kuvaruudun vyöhykkeiden (ZONE) leveydeksi 13 merkkipaikkaa eli saraketta. Kun PRINT-käskyyn sisältyy pilkku , saadaan seuraava merkki tai merkkiryhmä tulostumaan alkaen seuraavasta vyöhykepaikasta. Jos rivillä on jäljellä vähemmän sarakkeita eli merkkipaikkoja kuin mitä ZONE-käskyssä on määritelty, BASIC aloittaa seuraavan tulostettavan merkkiryhmän tulostuksen uudelta riviltä. Se ei siis katkaise tulostettavaa merkkiryhmää kesken rivin.

Jos PRINT-käskyn yhteydessä ei käytetä USING-muotoa, BASIC tulostaa positiiviset luvut siten, että niiden edessä on yksi tyhjä (välilyönti) ja negatiivisten lukujen edessä on miinusmerkki - . Kaikkien lukujen jäljessä on yksi tyhjä väli. BASIC jättää myös desimaalipisteen tulostamatta, ellei desimaalipisteen jälkeen ole mitään tulostettavaa.

[TAB]-näppäin ei toimi sarkainnäppäimenä AMSTRADin BASIC-kielessä, koska tämän näppäimen merkityksestä ja toiminnasta ei ole olemassa mitään yhdenmukaista käytäntöä eri BASIC-kielen murteissa. Ruutuun tulostuu oikealle päin oleva nuoli → painettaessa [TAB]-näppäintä (samoin käy, jos [CTRL]- ja I-näppäintä painetaan yhtäaikaa), mutta muuten [TAB]-näppäimellä ei ole sen suurempaa merkitystä AMSTRADin BASIC-kielessä.

3.7 PRINT TAB (<kokonaislukulauseke>) "tulostettava joukko"

Tämän käskymuodon vaikutus selviää parhaiten esimerkin avulla. Näppäile seuraava esimerkkiohjelma, ja katso kuinka se vaikuttaa:

```
5 MODE 2: INK 1,0: INK 0,9
10 FOR N=1 TO 5
20 ZONE 40
30 PRINT TAB(N*4)"HEI",N
40 NEXT
```

Tulostusvyöhykkeen leveydeksi on asetettu edellä olevassa ohjelmassa 40 saraketta eli merkkipaikkaa. Samalla ohjelmasta näkyy, kuinka pilkku ja TAB() toimivat käytännössä. Muuta rivi 10 seuraavaan muotoon ja aja ohjelma uudelleen:

```
10 FOR N=-5 to 5
```

TAB-käsky siirtää PRINT-käskyllä tulostettavan merkkijoukon alkukohtaa eteenpäin niin monta merkkipaikkaa kuin kohdassa <kokonaislukulauseke> on määritelty. (Tulostusvyöhykkeen leveydeksi voidaan määritellä ZONE-käskyllä 1 - 255 merkkipaikkaa - asiaa kuvataan tarkemmin luvussa 8).

PRINT USING-muotoa käytetään laskutoimitusten tulosten muotoilemiseen sellaisissa tapauksissa, joissa muuten desimaalipisteen jälkeen tulostuisi epäsäännöllinen määrä desimaalinumeroita. PRINT USING-lauseke on varsin monimutkainen, ja sen käyttö selviää parhaiten käytännön esimerkkien valossa, koska lausekkeen muotoa:

```
PRINT [#<tulostusreittilauseke>,][<tulostettava
joukko>][<USING-lauseke>][<erotin>]
```

ei voida pitää kovin käyttäjäystävällisenä varsinkaan kun USING-lauseke jakaantuu edelleen seuraaviin osiin:

```
USING <merkkijonolauseke>;[<USING-joukko>]
```

jossa USING-joukko jakaantuu edelleen seuraaviin osiin:

```
<lauseke>[<erotin><lauseke>]*
```

Asiat selviävät parhaiten, kun kokeilet seuraavaa ohjelmariviä:

```
PRINT 123.456, USING "###.##";4567.896
```

Tulos näkyy ruudussa seuraavasti:

```
123.456      %4567.90
```

Ruudussa näkyvästä tuloksesta näkyy montakin seikkaa. Ensiksi: käskyrivi ei ole vaikuttanut USING-lauseketta edeltävään osaan. Toiseksi: USING-lauseke varaa tietyn määrän tulostukseen käytettäviä merkkipaikkoja lainausmerkeissä määritellyllä tavalla sen jälkeen seuraavaa tulostettavaa osaa varten (joka voi tietenkin olla myös jokin muuttuja), ja jos tässä tulostettavassa osassa on enemmän merkkejä desimaalipisteen vasemmalla puolella kuin mitä sille on varattu, se kyllä tulostuu näyttöön, mutta sen eteen ilmestyy prosenttimerkki **%** ylityksen merkiksi. Kolmanneksi: numerolausekkeen 123.456 jälkeen oleva pilkku aiheutti sen, että seuraava luku tulostui alkaen seuraavan tulostusvyöhykkeen alkukohdasta. Jos pilkun tilalla olisi ollut puolipiste, olisi jälkimmäinen luku tulostunut heti numerolausekkeen 123.456 viereen yhden merkkipaikan päähän sen oikealle puolelle. Jos luvut tulostuvat samalle riville, niiden välissä on aina yksi tyhjä väli - syytä tuskin tarvitsee selittää.

Lopuksi kannattaa panna merkille, että tulostettava numerolauseke pyöristyy, eikä USING-lauseke suinkaan hylkää kylmästi viimeisen tulostukseen varatun merkkipaikan jälkeen olevia numeroita.

Kokeile seuraavaksi alla olevaa käskyriviä:

```
PRINT 123.456, USING "#####.##+";4567.896
```

Muotoillun luvun loppuun ilmestyy nyt plusmerkki. Negatiivisten lukujen perään ilmestyy miinusmerkki, ellei muuta ole määritelty.

PRINT USING-muoto on erittäin käyttökelpoinen tapa muotoilla esimerkiksi laskutoimitusten vastaukset eri sarakkeisiin. Lauseke antaa aina varoituksen prosenttimerkillä **%** , jos tulostukseen on varattu liian vähän merkkipaikkoja.

LUKU 4: MUUTTUJAT, OPERAATTORIT JA TIEDOT

Tässä luvussa käsitellään sitä, kuinka tietoja käsitellään BASIC-ohjelmassa. Käsiteltäviä aiheita ovat mm.:

- Erilaiset muuttujatyypit eli reaaliluku-, kokonaisluku- ja merkkijonomuuttujat
- Operaattorit ja loogiset operaatiot
- Taulukot (array)
- Tiedot

4.1 VARATTUJEN SANOJEN EROTTAMINEN

AMSTRADin BASIC-kieleen kuuluvat käskyt ja muut varatut sanat on erotettava toisistaan joko välilyönnillä, erotusmerkillä, laskutoimituksen merkillä, tai muulla vastaavalla merkillä. Kaikki avainsanat tulostuvat suurilla kirjaimilla pyydettäessä ohjelmalistaus LIST-käskyllä siitä riippumatta, onko avainsanat näppäilty pienillä kirjaimilla vaiko suurilla kirjaimilla. Näin ohjelmalistausten lukeminen on helpompaa ja niistä on helpompaa etsiä virheitä. Avainsanat kannattaa siis näppäillä pienillä kirjaimilla, sillä virhe paljastuu heti ohjelmalistauksesta, ellei aivansanat ole tulostuneet suurilla kirjaimilla. Ohjelma ei käynnisty, jos avainsana on näppäilty väärin.

AMSTRADin BASIC sallii avainsanojen käytön muuttujien nimissä. AMSTARDin BASICissa muuttuja tarkoittaa nimeä, joka annetaan tietylle alkiolle. Nimen ei tarvitse olla sen kummempi kuin pelkkä kirjain, mutta ohjelmalistausten lukeminen ja ymmärtäminen on paljon helpompaa, jos muuttujan nimenä käytetään kulloinkin suorituksen kohteena olevaan tehtävään liittyvää pitempää nimeä. Muuttujan nimen on aina alettava kirjaimella, eikä se saa koskaan alkaa numerolla. Kokeile seuraavaa:

```
VASTAUS=4*4: print VASTAUS
```

AMSTRADin BASIC hyväksyy jopa 40 merkkiä käsittäviä muuttujia (ensimmäisen merkin on aina oltava kirjain), ja kaikki 40 merkkiä ovat merkitseviä. Muuttujien nimissä ei saa olla välilyöntejä, sillä muuten BASIC lukee merkit (kirjaimet ja/tai numerot) ensimmäiseen välilyöntiin asti ja tulostaa sitten ruutuun muotovirheestä ilmoittavan virheilmoituksen **Syntax error** .

Tämä ilmoitus merkitsee siis sitä, että muuttujan nimeen on näppäilty sellainen merkkijono, jota AMSTRADin BASIC ei hyväksy (katso liitettä VIII). Jos haluat käyttää muuttujan nimenä lausetta jossa on käytetty toisistaan erotettuja sanoja, laita sanojen väliin piste siihen kohtaan johon muuten laittaisit välilyönnin. Indeksoituja muuttujia voidaan käyttää kaikissa tavallisissa muodoissaan, kunhan vain taulukot (array) muistetaan mitoittaa DIM-käskyllä.

4.2 OIKOPOLKUJA

Sinun ei suinkaan tarvitse näppäillä joka kerta PRINT vaan voit käyttää sen sijaan kysymysmerkkiä **?** , koska BASIC ymmärtää myös tämän PRINT-käskynä (kunhan sitä ei ole laitettu lainausmerkkien **""** sisään sijoitettuun lauseeseen). Kysymysmerkkiä ei tarvitse erottaa välilyönnillä samalla tavalla kuin PRINT-sanaa. Jos kirjoitat seuraavan käskyrivin:

```
10?4*4
run
```

niin ruutuun tulostuva vastaus on sama kuin jos antaisit käskyn suoraan ilman rivinumeroa, mutta jos pyydät listauksen edellä olevasta yksirivisestä ohjelmasta LIST-käskyllä, tapahtuu seuraavaa:

```
list
10 PRINT 4*4
```

BASIC on siis muuttanut kysymysmerkin PRINT-käskyksi, mutta sen lisäksi se on laittanut välilyönnin PRINT-sanan molemmin puolin. PRINT-lausekkeessa, jossa käytetään lainausmerkkejä seuraavaan tapaan:

```
10?"TERVEHDYS"
```

voidaan jälkimmäinen lainausmerkki jättää pois. Olet jo ehkä huomannutkin, että kun lataat kasetilta ohjelmaa tietokoneen muistiin painamalla [CTRL]-näppäintä ja pientä [ENTER]-näppäintä, tulostuu ruutuun **RUN"** . Myös ohjelmariveillä voidaan toimia samalla tavalla, mutta sitä ei kuitenkaan kannata ottaa tavaksi, koska jos myöhemmin palataan samalle riville ja tehdään muutoksia tai lisäyksiä kyseiselle riville, unohtuu jälkimmäinen lainausmerkki pois mitä ilmeisimmin.

4.3 USEAMPIA KÄSKYLAUSEKKEITA SISÄLTÄVÄT OHJELMARIVIT JA USEAMPIEN LASKUTOIMITUSTEN YHDISTÄMINEN

Voit ohjelmoida yhdelle BASIC-riville useammastakin käskystä koostuvia toimintoja. Ainoa rajoitus on se, ettei käskyrivin merkkimäärä voi olla suurempi kuin 255. Kaksoispistettä **:** käytetään käskylausekkeiden erottamiseen tavalliseen tapaan.

Jos näppäilet seuraavan käskyrivin:

```
?2*8/5+5-4*777E9/3
```

niin saat ruutuun vastauksen:

```
-1.036E+12
```

On kuitenkin tärkeää, että ymmärrät sen järjestyksen jossa BASIC suorittaa eri laskutoimitukset (+ - * / MOD jne.) sillä muuten et voi välttyä aivan perustavaa laatua olevilta virheiltä. Eri laskutoimitusten suoritusjärjestys ja niissä käytettävät merkit ovat:

↑ Potenssiinkorotus eli jokin luku korotetaan tiettyyn kymmenen potenssiin.
- Miinusmerkki (käytetään osoittamaan, että luku on negatiivinen).
* Kertolasku.
/ Jakolasku.
\ Jakolasku siten, että tulokseen otetaan mukaan vain kokonaisosa ja mahdollinen desimaaliosa jätetään pois.
+ Yhteenlasku.
- Vähennyslasku.

BASIC käsittelee aivan ensimmäiseksi kaiken sen, mitä sulkumerkkien **( )** sisään on laitettu. Jos sulkumerkkien sisällä on jo useammassa laskutoimituksesta koostuva tehtävä, käsittelee BASIC tehtävät edellä luetellussa järjestyksessä. Samalla BASIC huomioi myös sulkumerkkien sisällä mahdollisesti olevien muiden sulkumerkkien aiheuttaman laskujärjestyksen. Muista tarkistaa, että käskyrivillä on aina yhtä monta avautuvaa sulkumerkkiä **(** kuin sulkeutuvaa sulkumerkkiäkin **)** , koska muuten ruutuun tulostuu muotovirheestä ilmoittava virheilmoitus **Syntax error** .

4.4 MENNÄÄNPÄ ETEENPÄIN!

Olemme kahlanneet läpi jo koko joukon asioita sen jälkeen kun aloitimme tutustumisen PRINT-lausekkeeseen kohdassa 3.5. Olet varmaankin jo oppinut riittävästi AMSTRADin BASICin perussääntöjä, jotta voimme siirtyä eteenpäin ja aloittaa tutustumisen BASIC-kielen käyttöalueisiin, jotka ovat enemmän tunnusomaisia "oikeille" tietokoneille kuin yksinkertaisille laskimille. BASIC-kielen avainsanoja käsitellään aina tarpeen mukaan, ja niiden merkityksen voi tarkistaa luvusta 8, ellei avainsanojen merkitys selviä asiayhteydestä.

BASIC-kielen avainsanat perustuvat englanninkielisiin lyhenteisiin, ja niiden merkitys selviää usein jo itse avainsanasta. Esimerkiksi GOTO 50 tarkoittaa: mene riville numero 50 ja jatka ohjelman suoritusta siitä. END merkitsee myös sananmukaisesti loppua, ja kuvaruutuun tulostuu viesti **Ready** aina kun ohjelma tulee END-käskyn kohdalle vaikka se olisi heti ohjelman ensimmäisellä rivillä.

Voit syöttää tietokoneeseen suoraan näppäimistöltä useampiakin käskyjä peräkkäin, kunhan vain muistat laittaa käskylausekkeiden väliin kaksoispisteen **:** . Tällöin käy kuitenkin niin, että kun käsket tietokoneen suorittamaan ohjelmarivin painamalla [ENTER]-näppäintä, tietokone käsittelee rivillä olevat käskyt ja unohtaa sitten rivin. Voit tietenkin ajaa rivillä olevat käskyt uudelleen käyttämällä hyväksesi kopiointikohdistinta - tietenkin sillä edellytyksellä, että käskyrivi jää näkyviin kuvaruutuun sen jälkeen kun tietokone on suorittanut käskyrivin.

4.5 EHDOLLISET JA LOOGISET LAUSEKKEET

Tietokone pystyy tunnetusti suorittamaan toistuvia tehtäviä varsin suurella nopeudella väsymättä ja kyllästymättä pitkiinkään tehtäväjaksoihin. BASIC pystyy hyödyntämään juuri näitä tietokoneen ominaisuuksia varsin monipuolisesti. BASIC-kielessä on runsaasti ohjelmointikäskyjä, joita käytetään toistuvien tehtävien eli silmukoiden muodostamiseen. Käskyjä käytetään aloittamaan, jatkamaan ja päättämään silmukat ennalta määriteltyjen ehtojen täyttyessä.

Silmukan käyttäytymisen ohjaukseen käytetään myös vertailulausekkeita. Niissä tutkitaan, minkälainen jokin tieto on toiseen tietoon verrattuna. Vertailu voidaan tehdä esimerkiksi kahden muuttuvan tiedon välillä tai muuttuvaa tietoa voidaan verrata johonkin ennalta määriteltyyn (kiinteään) vertailuarvoon. Vertailu tapahtuu "vertailuoperaattoreilla", joita ovat:

< pienempi kuin
<= pienempi tai yhtäsuuri kuin
= yhtäsuuri kuin
> suurempi kuin
>= suurempi tai yhtäsuuri kuin
<> erisuuri kuin

Seuraavana meillä onkin lyhyt ohjelma, joka käsittelee meidän kaikkien sydäntä lähellä olevaa asiaa. Tutustumme tässä ja seuraavissa ohjelmissa vertailuoperaattoreiden käyttöön. Paina [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä tässä järjestyksessä pitäen ne yhtäaikaa alhaalla, jolloin kuvaruutuun ilmestyy alkuviesti eli nollausviesti. Näppäile sitten koneeseen seuraava ohjelma (jos teet virheen näppäillessäsi ohjelmaa, löydät korjailuohjeet kohdasta 1.2.7):

```
10 INPUT "PALJONKO SAAT PALKKAA KUUKAUDE
SSA";PALKKA
20 IF PALKKA <5000 THEN GOTO 30 ELSE 40
30 PRINT "KYSY PALKANKOROTUSTA":END
40 PRINT "KOITA SAADA SUUREMPI AUTO"
```

Aja ohjelma RUN-käskyllä (ja muista painaa [ENTER]-näppäintä). Kuvaruudussa näkyy nyt kysymys:

```
PALJONKO SAAT PALKKAA KUUKAUDESSA?
```

(Tietokone lisää automaattisesti kysymysmerkin kun se haluaa, että näppäimistöltä syötetään tietoja vastauksena INPUT-lausekkeella esitettyyn kysymykseen). Vastaa kysymykseen pelkillä numeroilla, äläkä käytä vastauksessa mitään kirjaimia, merkkejä tai pilkkuja. Syötä vastaus koneeseen painamalla [ENTER]-näppäintä.

Lisää ohjelmaan alla oleva rivi 5 näppäilemällä se sen jälkeen kun kuvaruutuun on ilmestynyt **Ready** -ilmoitus ohjelman suorituksen jälkeen.

```
5 CLS
```

Tyhjennä ruutu ajamalla ohjelma uudelleen RUN-käskyllä. Jos ensimmäinen vastauksesi oli alle 5000, vastaa nyt suuremmalla summalla kuin 5000, jolloin näet erilaisen vastauksen. Lisää sitten rivi 50 alla olevan esimerkin mukaisesti:

```
5 CLS
10 INPUT "PALJONKO SAAT PALKKAA KUUKAUDE
SSA";PALKKA
20 IF PALKKA <5000 THEN GOTO 30 ELSE 40
30 PRINT "KYSY PALKANKOROTUSTA":END
40 PRINT "KOITA SAADA SUUREMPI AUTO"
50 IF PALKKA >10000 THEN PRINT "ja hanki
 itsellesi taitava kirjanpitaja"
run
```

Rivin 30 lopussa oleva END-käsky pysäyttää ohjelman ja aiheuttaa **Ready** -ilmoituksen tulostumisen ruutuun. Rivin 40 loppuun ei ole laitettu END-käskyä, joten ohjelma jatkaa tarkistaakseen, onko PALKKA-muuttuja suurempi kuin 10000. Jos se on suurempi kuin 10000, ohjelma tulostaa ruutuun varsin varteenotettavan neuvon.

Lisää ohjelmaan vielä rivi 60:

```
5 CLS
10 INPUT "PALJONKO SAAT PALKKAA KUUKAUDE
SSA";PALKKA
20 IF PALKKA <5000 THEN GOTO 30 ELSE 40
30 PRINT "KYSY PALKANKOROTUSTA":END
40 PRINT "KOITA SAADA SUUREMPI AUTO"
50 IF PALKKA >10000 THEN PRINT "ja hanki
 itsellesi taitava kirjanpitaja"
60 IF PALKKA >8000 THEN PRINT "... ja vi
ppaa minulle viitonen"
run
```

Vertailuoperaattorit > ja < toimivat rajoittimina merkiten luvun rajakohdat. Merkkien eteen tai jälkeen ei tarvitse laittaa välilyöntiä. Jos laitat välilyönnin, saattaa BASIC hylätä välilyönnin joissakin tapauksissa. Kun ajat ohjelman ja vastaat palkaksesi yli 8000 mutta alle 10000, BASIC ohittaa rivin 50 aivan kuin sitä ei olisi koko ohjelmassa ja suorittaa rivillä 60 annetun tehtävän.

Tässä vaiheessa voisit tehdä aivan oman ohjelman käyttäen siinä tähän mennessä käsiteltyjä asioita. Huomasit varmaan, kuinka esimerkkiohjelma "kasvoi" koko ajan. Useimpia ohjelmia tehtäessä käy juuri näin. Tämä kuvaa ehkä parhaiten erästä BASIC-ohjelmoinnin tärkeimmistä perusasioista, eli ohjelma voidaan muokata koska hyvänsä uudelleen aina paremmaksi ja monipuolisemmaksi.

4.6 OHJELMIEN KEHITTÄMINEN

BASIC-ohjelmassa on kätevintä juuri se, kuinka sitä pystytään kehittämään ja täydentämään sitä mukaa kun ideoita ja ohjelmointikykyä karttuu. Tämä on joidenkin mielestä liian leväperäinen tapa tehdä ohjelmia, ja se johtaa heidän mielestään sattumanvaraisiin ideoihin perustuvaan ohjelmointitekniikkaan. Toiset taas ovat sitä mieltä, että tämä on juuri BASIC-kielen paras puoli, koska ohjelman laatija pystyy tarkistamaan ohjelmansa missä vaiheessa hyvänsä ja täydentämään sitä niin halutessaan.

Ottakaamme esimerkkiohjelmamme jälleen käsittelyn kohteeksi ja lisätkäämme siihen rivi 70. Siinä on silmukka, joka ohjaa ohjelman suorituksen takaisin alkuun pidettyään riittävän pitkän tauon, jotta näytössä olevat tekstit ehtii helposti lukea:

```
5 CLS
10 INPUT "PALJONKO SAAT PALKKAA KUUKAUDE
SSA";PALKKA
20 IF PALKKA <5000 THEN GOTO 30 ELSE 40
30 PRINT "KYSY PALKANKOROTUSTA":END
40 PRINT "KOITA SAADA SUUREMPI AUTO"
50 IF PALKKA >10000 THEN PRINT "ja hanki
 itsellesi taitava kirjanpitaja"
60 IF PALKKA >8000 THEN PRINT "... ja vi
ppaa minulle viitonen"
70 for n=1 to 3500: next n: goto 5
run
```

Uusi rivi samoin kuin muutkin alkuperäiseen ohjelmaan lisätyt rivit on näppäilty pienillä kirjaimilla. Tämän tarkoituksena on muistuttaa, että AMSTRADin BASIC pystyy erottamaan, koska kysymyksessä on muuttujan nimi ja koska avainsana. Poistu ohjelmasta painamalla [ESC]-näppäintä kahdesti ja pyydä ohjelmalistaus LIST-käskyllä. Näet, kuinka tietokone muuttaa avainsanat tulostaen ne suurilla kirjaimilla mutta muuttuja n tulostuu edelleen pienellä.

Rivin 70 alussa on viivästyssilmukka, jossa tietokone antaa muuttujalle n arvot 1 - 3500 ennen kuin se suorittaa seuraavan lausekkeen GOTO 5. Ohjelma kiertää päättymättömässä silmukassa, ellet vastaa tietokoneen esittämään kysymykseen, että saat palkkaa vähemmän kuin 5000 mk kuukaudessa. Muussa tapauksessa ohjelmasta pääsee ulos vain painamalla [ESC]-näppäintä. Ensimmäinen painallus keskeyttää ohjelman suorituksen ja toinen painallus katkaisee ohjelman. Tällöin kuvaruutuun ilmestyy **Ready** -ilmoitus, jolloin tietokone on valmis vastaanottamaan uusia käskyjä. Ohjelman katkaiseminen [ESC]-näppäimellä ei hävitä ohjelmaa muistista.

Oikeastaan ohjelma katkeaa välittömästi painaessasi [ESC]-näppäintä, ellet paina sitä juuri silloin, kun tietokone suorittaa rivillä 70 olevaa viivästyssilmukkaa. Näin käy siksi, että tietokone ei ole suorittamassa mitään tehtävää odottaessaan näppäimistöltä syötettäviä palkkatietoja. **Break** -sanan jälkeen tulostuu se rivinumero, jolla tietokone oli odottamassa katkaisun tapahtuessa. Kuvaruutuun ilmestyy seuraava viesti, jos painat [ESC]-näppäintä tietokoneen odottaessa näppäimistöltä syötettäviä palkkatietoja:

```
Break in 10
```

Ruutuun ilmestyvä viesti on, puolestaan seuraava, jos katkaiset ohjelman rivillä 70 olevan viivästyssilmukan aikana:

```
Break in 70
```

Tietokone jatkaa ohjelman suorittamista painaessasi mitä hyvänsä näppäintä, jos keskeytät ohjelman rivillä 70 olevan silmukan aikana painamalla [ESC]-näppäintä yhden kerran. Voit jatkaa ohjelman suoritusta myös katkaistuasi ohjelman painamalla [ESC]-näppäintä kahdesti. Tämä tapahtuu antamalla CONT-käsky. Tällöin ohjelma jatkuu siltä riviltä, jolla ohjelma keskeytettiin.

Ohjelma ei siis häviä ohjelmamuistista, käytätpä [ESC]-näppäintä miten hyvänsä. Ohjelma häviää muistista vasta sitten, kun annat tietokoneelle muistintyhjennyskäskyn NEW tai nollaat tietokoneen aivan kokonaan painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä yhtäaikaa.

Mitään sen kummempia varmuustoimenpiteitä ei tarvita, koska muistin tyhjentäminen vahingossa ei ole mahdollista. Ohjelmamuistin tyhjentäminen edellä mainituilla toimenpiteillä on siis aina harkittu teko - ja peruuttamaton. Siksi kasetille kannattaakin aina tallentaa kaikki sellaiset ohjelmat, joita ehkä haluat käyttää myöhemmin uudelleen.

4.7 LISÄÄ MUUTTUJIA JA MERKKIJONOJA

Muuttujat kuuluvat tietokoneisiin aivan oleellisesti. Tietokone olisi vain pelkkä elektroninen lasku- ja kirjoituskone, jos se käsittelisi pelkästään kiinteitä lukuja tai merkkijonoja. Tässä yhteydessä on syytä huomauttaa, että jos matemaattiseen lausekkeeseen kuuluu muuttuja, on myös tulos muuttuja.

Muuttujilla on kolme ominaisuutta: nimi, tyyppi ja "organisaatio". Nimiä käsiteltiin jo aikaisemmin (4.1) ja tyypit ovat valinnaisia, joten voisimme määritellä muuttujan kohdassa 3.4 olevien sääntöjen mukaan seuraavasti:

<nimi>[<tyyppiä osoittava merkki>]

Tyyppiä osoittavia merkkejä on kolme. Prosenttimerkkiä % käytetään kokonaisluvuille, jolloin desimaalipisteen oikealla puolella olevaa desimaaliosaa ei huomioida. Kokonaisluvun muodossa olevat muuttujat tarvitsevat vähemmän muistitilaa, jolloin sellaiset ohjelmat, joissa tullaan toimeen ilman lukujen desimaaliosaa, saadaan toimimaan nopeammin määriteltäessä muuttujat kokonaisluvuiksi. Muuttujat määritellään kokonaisluvuiksi DEFINT-käskyllä. Kokonaislukualue on -32768 ... +32767.

Huutomerkki ! ilmoittaa, että muuttuja on reaaliluku. Tämä puolestaan tarkoittaa, että muuttujassa on kokonaisosa ja desimaalipisteen oikealla puolella oleva desimaaliosa. Tietokone olettaa, että muuttujat ovat reaalilukuja kytkettäessä virta tietokoneeseen. Muuttujia ei siis tarvitse määritellä reaaliluvuiksi muulloin kuin jos olet aikaisemmin käyttänyt DEFINT-käskyä. Reaalilukumuuttujien arvo voi olla väliltä 2.9E-39 ... 1.7E+38.

Dollarinmerkki $ osoittaa, että muuttuja on merkkijonomuuttuja. Tällaisessa muuttujassa voi olla numeroita, kirjaimia ja muita merkkejä. Merkkijonomuuttuja voi koostua siis lainausmerkkeihin "" sijoitetusta mielivaltaisesta merkkivalikoimasta. Esimerkki merkkijonomuuttujasta:

```
NIMI$="JASKA JOKUNEN"
```

Käytämme esimerkkiohjelmaa apunamme ja lisäämme siihen rivin 6 sekä korjaamme rivin 60 seuraavasti:

```
5 CLS
6 INPUT "Kuka olet (kirjoita nimesi)";NI
MI$
10 INPUT "PALJONKO SAAT PALKKAA KUUKAUDE
SSA";PALKKA
20 IF PALKKA <5000 THEN GOTO 30 ELSE 40
30 PRINT "KYSY PALKANKOROTUSTA":END
40 PRINT "KOITA SAADA SUUREMPI AUTO"
50 IF PALKKA >10000 THEN PRINT "ja hanki
 itsellesi taitava kirjanpitaja"
60 IF PALKKA >8000 THEN PRINT "... ja vi
ppaa minulle viitonen ";NIMI$
70 for n=1 to 3500: next n: goto 5
run
```

Sanan **viitonen** jälkeen on lisätty yksi välilyönti, koska muuten nimi tulostuisi heti sen perään ilman väliä. Kokeile ellet usko! PRINT- tai INPUT-lausekkeen perässä oleva puolipiste ; estää tietokonetta aloittamasta uutta riviä heti lausekkeen jälkeen, ellei sitä erikseen käsketä niin tekemään.

Voimme myös kokeilla kokonaislukumuuttujia. Näppäile rivi 61 alla esitetyssä muodossa, jolloin ohjelmalistaus näyttää seuraavalta:

```
5 CLS
6 INPUT "Kuka olet (kirjoita nimesi)";NI
MI$
10 INPUT "PALJONKO SAAT PALKKAA KUUKAUDE
SSA";PALKKA
20 IF PALKKA <5000 THEN GOTO 30 ELSE 40
30 PRINT "KYSY PALKANKOROTUSTA":END
40 PRINT "KOITA SAADA SUUREMPI AUTO"
50 IF PALKKA >10000 THEN PRINT "ja hanki
 itsellesi taitava kirjanpitaja"
60 IF PALKKA >8000 THEN PRINT "... ja vi
ppaa minulle viitonen ";NIMI$
61 PAIVAPALKKA=PALKKA*12/365: PRINT "Pai
vapalkaksi tulee";PAIVAPALKKA;" mk paiva
ssa"
70 for n=1 to 6000: next n: goto 5
run
```

Rivillä 70 olevaa viivästyssilmukkaa on kasvatettu 6000:een, koska ruudussa on enemmän luettavaa. Päiväpalkan laskutulos on hieman epäsiistin näköinen, joten se voidaan aivan hyvin pyöristää kokonaisluvuksi lisäämällä ohjelmaan rivi 62.

```
5 CLS
6 INPUT "Kuka olet (kirjoita nimesi)";NI
MI$
10 INPUT "PALJONKO SAAT PALKKAA KUUKAUDE
SSA";PALKKA
20 IF PALKKA <5000 THEN GOTO 30 ELSE 40
30 PRINT "KYSY PALKANKOROTUSTA":END
40 PRINT "KOITA SAADA SUUREMPI AUTO"
50 IF PALKKA >10000 THEN PRINT "ja hanki
 itsellesi taitava kirjanpitaja"
60 IF PALKKA >8000 THEN PRINT "... ja vi
ppaa minulle viitonen ";NIMI$
61 PAIVAPALKKA=PALKKA*12/365: PRINT "Pai
vapalkaksi tulee";PAIVAPALKKA;" mk paiva
ssa"
62 KOKONAISET.MARKAT%=PAIVAPALKKA: PRINT
 "eli";KOKONAISET.MARKAT%;"markkaa jos p
ennit unohdetaan"
70 FOR n=1 TO 6000: NEXT n: GOTO 5
run
```

Muuttujan tyyppiä osoittavana merkkinä on muistettava käyttää prosenttimerkkiä %, koska laskutoimituksen tulokseksi saattaa tulla reaalilukumuuttuja, jonka lukuarvo on sama kuin kokonaislukumuuttujallakin. Desimaaliosa saadaan siis jäämään pois käyttämällä prosenttimerkkiä osoittamassa muuttujan tyyppiä. Samalla näet ohjelmalistauksesta, kuinka tietokone katkaisee ohjelmarivin ja jatkaa sitä seuraavalta riviltä, ellei se mahdu yhdelle riville (näin käy kaikissa kolmessa näyttömuodossa). Pitkiä ohjelmia kirjoitettaessa on helpompaa käyttää näyttömuotoa 2 (mode 2), koska ohjelmalistauksia on paljon helpompi lukea "ehjiltä" ohjelmariveiltä. Pääset näyttömuotoon 2 näppäilemällä MODE 2.

Näppäile seuraavat kolme käskyä suoraan näppäimistöltä, jos haluat käyttää mustia merkkejä valkoisella pohjalla:

```
INK 1,0
INK 0,13
BORDER 13
```

Pyydä nyt ohjelmalistaus ruutuun LIST-käskyllä.

4.8 NÄYTÖN MUOTOILU

Ohjelman kehittelyyn kuuluu oleellisesti, että sitä siistitään aina silloin tällöin. Aluksi voimme numeroida kaikki rivit uudelleen kymmenen välein käyttämällä RENUM-käskyä. Näppäile siis RENUM-käsky, kun kuvaruudussa näkyy **Ready** -ilmoitus. Pyydä tämän jälkeen ohjelmalistaus LIST-käskyllä, jolloin ohjelmalistaus näyttää näyttömuodossa 1 (Mode 1) seuraavalta:

```
10 CLS
20 INPUT "Kuka olet (kirjoita nimesi)";N
IMI$
30 INPUT "PALJONKO SAAT PALKKAA KUUKAUDE
SSA";PALKKA
40 IF PALKKA <5000 THEN GOTO 50 ELSE 60
50 PRINT "KYSY PALKANKOROTUSTA":END
60 PRINT "KOITA SAADA SUUREMPI AUTO"
70 IF PALKKA >10000 THEN PRINT "ja hanki
 itsellesi taitava kirjanpitaja"
80 IF PALKKA >8000 THEN PRINT "... ja vi
ppaa minulle viitonen ";NIMI$
90 PAIVAPALKKA=PALKKA*12/365: PRINT "Pai
vapalkaksi tulee";PAIVAPALKKA;" mk paiva
ssa"
100 KOKONAISET.MARKAT%=PAIVAPALKKA: PRIN
T "eli";KOKONAISET.MARKAT%;"markkaa jos
pennit unohdetaan"
110 FOR n=1 TO 6000: NEXT n: GOTO 10
run
```

Kaikki rivinumerot ovat pyöristyneet ylöspäin - myös ne, joihin viitataan ohjelman sisällä. Eipä siitä olisikaan juuri hyötyä, ellei BASIC pitäisi huolta kaikista rivinumeroista ja päivittäisi niitä yhtäaikaa. Seuraavaksi siistimme näyttöä. Tämä tapahtuu estämällä aluksi rivillä 110 olevan silmukan toiminta muuttamalla käskyrivi huomautusriviksi REM-käskyllä:

```
10 CLS
20 INPUT "Kuka olet (kirjoita nimesi)";N
IMI$
30 INPUT "PALJONKO SAAT PALKKAA KUUKAUDE
SSA";PALKKA
40 IF PALKKA <5000 THEN GOTO 50 ELSE 60
50 PRINT "KYSY PALKANKOROTUSTA":END
60 PRINT "KOITA SAADA SUUREMPI AUTO"
70 IF PALKKA >10000 THEN PRINT "ja hanki
 itsellesi taitava kirjanpitaja"
80 IF PALKKA >8000 THEN PRINT "... ja vi
ppaa minulle viitonen ";NIMI$
90 PAIVAPALKKA=PALKKA*12/365: PRINT "Pai
vapalkaksi tulee";PAIVAPALKKA;" mk paiva
ssa"
100 KOKONAISET.MARKAT%=PAIVAPALKKA: PRIN
T "eli";KOKONAISET.MARKAT%;"markkaa jos
pennit unohdetaan"
110 REM:FOR n=1 TO 6000: NEXT n: GOTO 10
run
```

REM-käskyn lisääminen rivin 110 alkuun aiheuttaa sen, että BASIC ei enää otakaan huomioon sen jälkeen olevia käskyjä vaan päättää ohjelman ja tulostaa ruutuun **Ready** -ilmoituksen. Samalla tekstit jäävät näkyviin kuvaruutuun. Näppäile nyt seuraava rivi:

```
15 mode 1
```

Rivi 15 aiheuttaa nyt sen, että näyttömuodoksi tulee ohjelman alussa näyttömuoto 1 siitä riippumatta, mikä näyttömuoto oli käytössä käynnistäessäsi ohjelman RUN-käskyllä. MODE-käsky suorittaa myös automaattisesti kuvaruudun tyhjennyksen (CLS), joten rivi 10 on nyt turha, mutta jätämme sen kuitenkin paikalleen.

Käynnistä ohjelma RUN-käskyllä ja vastaa kysymyksiin alla esitetyllä tavalla:

```
Kuka olet (kirjoita nimesi)? Jaska
PALJONKO SAAT PALKKAA KUUKAUDESSA? 11000

KOITA SAADA SUUREMPI AUTO
ja hanki itsellesi taitava kirjanpitaja
... ja vippaa minulle viitonen Jaska
Paivapalkaksi tulee 361.643836
 mk paivassa
eli 362 markkaa jos pennit unohdetaan
Ready
```

Siisti ohjelmaa lisäämällä siihen rivit:

```
25 PRINT
85 PRINT
```

Siirrä **Ready** -ilmoitusta alemmas kuvaruudussa lisäämällä rivi 120:

```
120 ?:?:?:?:?
```

Voit ajaa ohjelman tällaisenaan tai poistaa **Ready** -ilmoituksen kokonaan näppäilemällä riviksi 120:

```
120 GOTO 120
```

Voit poistua tästä ohjelmasta käynnistettyäsi sen RUN-käskyllä ainoastaan painamalla [ESC]-näppäintä. Muistathan, että kysymysmerkki **?** on lyhennetty muoto PRINT-käskystä. Pyydä nyt ohjelmalistaus LIST-käskyllä ja katso mitä olet saanut aikaan tähän mennessä:

```
10 CLS
15 MODE 1
20 INPUT "Kuka olet (kirjoita nimesi)";N
IMI$
25 PRINT
30 INPUT "PALJONKO SAAT PALKKAA KUUKAUDE
SSA";PALKKA
40 IF PALKKA <5000 THEN GOTO 50 ELSE 60
50 PRINT "KYSY PALKANKOROTUSTA":END
60 PRINT "KOITA SAADA SUUREMPI AUTO"
70 IF PALKKA >10000 THEN PRINT "ja hanki
 itsellesi taitava kirjanpitaja"
80 IF PALKKA >8000 THEN PRINT "... ja vi
ppaa minulle viitonen ";NIMI$
85 PRINT
90 PAIVAPALKKA=PALKKA*12/365: PRINT "Pai
vapalkaksi tulee";PAIVAPALKKA;" mk paiva
ssa"
100 KOKONAISET.MARKAT%=PAIVAPALKKA: PRIN
T "eli";KOKONAISET.MARKAT%;"markkaa jos
pennit unohdetaan"
110 REM:FOR n=1 TO 6000: NEXT n: GOTO 10
120 GOTO 120
```

4.9 KOHDISTIMEN SIJOITUS LOCATE-KÄSKYLLÄ

Tähän mennessä useimmat käyttämistämme BASIC-käskyistä ovat olleet sellaisia, jotka kuuluvat useimpien muidenkin mikrotietokoneiden BASIC-käskyvalikoimaan. LOCATE-käsky on kuitenkin eräs sellainen käsky, joka on tyypillinen AMSTRADin BASIC-kielelle (sama käsky löytyy useista muistakin BASIC-murteista). Tekstikohdistin voidaan sijoittaa mihin hyvänsä kuvaruudun kohtaan LOCATE-käskyllä. Esimerkki:

```
LOCATE 10,4
```

Tämä käsky sijoittaa tekstikohdistimen kymmenenteen sarakkeeseen neljännelle riville kuvaruudun yläreunasta laskien. Jos yrität näppäillä tämän suoraan näppäimistöltä, kohdistin kylläkin sijoittuu määriteltyyn kohtaan mutta **Ready** -ilmoitus aiheuttaa sen, että tietokone aloittaa uuden rivin, jolloin kohdistin siirtyy taas vasempaan reunaan. Lisää äskeiseen palkkaohjelmaan rivi 16:

```
16 LOCATE 1,4
run
```

Ensimmäinen kysymysrivi on nyt alentunut neljännelle riville. Paina [ESC]-näppäintä kahdesti ja lisää vielä rivi 26:

```
26 LOCATE 1,6
run
```

Voit siis sijoittaa ohjelman tekstit tarkasti siihen ruudun kohtaan, johon itse haluat. Voit myös määritellä LOCATE-käskyllä saman rivin, jolloin uusi kysymys kirjoittuu vanhan kysymyksen päälle. Liitteessä VI on lueteltu koordinaatit, joista voi olla apua tekstien suunnittelussa.

LOCATE-käskyn eteen on laitettava CLS-käsky, jos haluat kaikkien kysymysten tai kommenttien tulostuvan samalle riville. Tällä tavalla vanhan tekstin rippeet eivät jää häiritsemään kuvaruutuun. LOCATE-käskyn koordinaatit voivat myös olla ohjelmassa määriteltyjä muuttujia.

Lopetamme nyt palkkaohjelman käsittelyn ja siirrymme seuraavaan aiheeseen. Voit tallentaa ohjelman kasetille luvussa 2 olevien ohjeiden mukaisesti. Tällä tavalla voit laajentaa ja monipuolistaa ohjelmaasi sitä mukaa kun uusia käskyjä tulee vastaan myöhemmässä vaiheessa.

4.10 IF...THEN

Tämän käsky-yhdistelmän eräs yleisimmistä muodoista on IF <looginen lauseke> THEN GOTO <rivinumero>.

IF-käskyllä tarkistetaan, onko loogisen lausekkeen tulos tosi. Jos näin on, ohjelma suorittaa sen perässä olevan lausekkeen. IF-käsky voidaan myös laittaa osaksi itseään toistavia silmukoita. Nollaa tietokone painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä ja näppäile:

```
1 MODE 1
10 AMSTRAD=0
20 PRINT "AMSTRAD CPC464 colour persona
l computer"
30 AMSTRAD=AMSTRAD +1
40 IF AMSTRAD <10 GOTO 20
run
```

Kuten näit, rivillä 20 oleva lause tulostuu niin monta kertaa kunnes rivillä 40 oleva ehto täyttyy. Silmukka on siis kytketty riviltä 40 riville 20. AMSTRAD-sanaa on käytetty muuttujana, jonka arvo muuttuu jokaisen silmukan aikana.

Voimme lisätä rivin 35 nähdäksemme, mitä AMSTRAD-muuttujan arvolle tapahtuu ohjelman aikana:

```
35 LOCATE 1,20: PRINT AMSTRAD:LOCATE 1,
AMSTRAD
```

Lisää ohjelmaan alla oleva viivästyssilmukka, jos et kerinnyt näkemään mitä kuvaruudussa tapahtui:

```
36 for n=1 to 500: next
```

Saat ohjelman hieman värikkäämmäksi (jos laitteistoosi kuuluu monitori CTM 640) lisäämällä seuraavan rivin:

```
34 BORDER AMSTRAD
```

Tämä rivi vaihtaa reuna-alueen (border) väriä rivillä 30 olevan AMSTRAD-muuttujan mukaan. Ohjelmalistaus näyttää nyt seuraavalta:

```
1 MODE 1
10 AMSTRAD=0
20 PRINT "AMSTRAD CPC464 colour persona
l computer"
30 AMSTRAD=AMSTRAD +1
34 BORDER AMSTRAD
35 LOCATE 1,20: PRINT AMSTRAD:LOCATE 1,
AMSTRAD
36 FOR n=1 to 500:NEXT
40 IF AMSTRAD <10 GOTO 20
run
```

Haluat varmaankin nähdä kaikki CPC 464:n värivalikoimaan kuuluvat värit, joten korjaa rivi 40 muotoon:

```
40 IF AMSTRAD <26 GOTO 20
```

Reuna-alueen väri vaihtuu nyt AMSTRAD-muuttujan arvoa vastaavasti alkaen tummimmasta ja päättyen kirkkaanvalkoiseen. Tee vielä yksi parannus ohjelmaan muuttamalla rivi 35 alla esitettyyn muotoon:

```
35 LOCATE 1,20: PRINT "Reuna-alue ";AMST
RAD: LOCATE 1,AMSTRAD
run
```

Kun nyt kerran olemme käsittelemässä reuna-aluetta, näppäile seuraava rivi kun kuvaruudussa näkyy jälleen **Ready** -ilmoitus:

```
BORDER 14,6
```

Reuna-alueen väri vaihtelee nyt värien 14 ja 6 välillä. Värien ja grafiikan toimintaa käsitellään tarkemmin seuraavassa luvussa. Näppäile vielä tämän jakson päätteeksi:

```
ink 1,18,16
```

ja vielä

```
speed ink 1,5
```

Olet jo varmaan päästäsi pyörällä ja tarvitset ehkä pari disperiiniä. Muista tallentaa ohjelma kasetille ennen kuin ohjelma häviää muistista.

4.11 TAULUKOT (ARRAY)

Joissakin ohjelmissa tarvitaan paljon muuttujia tietojen tallentamiseksi. Tämä käy kylläkin hyvin päinsä, mutta vaikeuksia alkaa ilmetä heti kun yritetään pysyä kärryillä siitä, mitä muuttujaa käytetään minkäkin tiedon tallentamiseen. Onneksi BASIC tarjoaa ratkaisun tilanteeseen. Ratkaisun nimi on taulukot.

Ihmettelet varmaan, mikä se sellainen taulukko oikein on. Se on periaatteessa ryhmä muuttujia, joihin kaikkiin viitataan samalla avainsanalla.

Asia selviää parhaiten käytännön esimerkin avulla. Ajatelkaamme korttipeliä jäljittelevää ohjelmaa, johon kuuluu oleellisesti sattumanvaraisesti valittujen korttien jakaminen. Samaa korttia ei ilmeisestikään voida jakaa kahdesti, joten jokaisesta yksittäisestä kortista on pidettävä kirjaa. Tämä voitaisiin tehdä yksinkertaisesti nimeämällä jokaiselle kortille muuttuja ja antamalla muuttujalle esimerkiksi arvo 1 jos kortti on jaettu ja arvo 0 jos kortti on edelleen pakassa.

Tämä edellyttää mitä ilmeisimmin 52 erillisen muuttujan käyttöä. Samalla sinun pitäisi muistaa, mikä muuttuja kuuluu millekin kortille. Juuri tällaisissa tilanteissa taulukot osoittautuvat varsin käyttökelpoisiksi.

Ensiksi taulukolle on annettava nimi - vaikkapa PAKKA. Jokaiselle taulukon alkiolle on annettava oma numeronsa, jotta saisimme kunkin alkion halutessamme käyttöömme. Jos vaikka kolmeatoista ensimmäistä alkiota käytetään maan - vaikkapa hertan - määrittelyyn, olisi herttakuutonen PAKKA(6) ja herttakymppi PAKKA(10) ja herttakuningas PAKKA(13). Alkaako valjeta?

Et voi kuitenkaan jatkaa tällä tavalla antamalla taulukon alkioille viitenumeroita loputtomiin, vaan sinun on annettava tietokoneelle selvä ilmoitus siitä, kuinka paljon sen on varattava tähän tarkoitukseen muistitilaa. Siihen käytetään DIM-käskyä.

DIM-käsky mitoittaa taulukon eli määrittelee sen koon eli ulottuvuuden. Näin siis korttipakkaesimerkissämme on varattava 52 alkiota. BASIC-kielellä tämä tarkoittaa käskyriviä:

```
DIM PAKKA(52)
```

Tällä käskyllä tietokone saadaan varaamaan muistitilaa 52 muuttujaa varten (itse asiassa 53 muuttujaa varten, koska myös alkioon nro 0 päästään käsiksi).

Seuraavaksi teemme kortinjakoon käytettävän aliohjelman rungon:

```
10 DIM PAKKA(52)
20 FOR X = 1 TO 52
30 LET PAKKA(X)=0
40 NEXT X
......
1000 KORTTI = RND(52)+1
1010 IF PAKKA(KORTTI) =1 THEN GOTO 1000
1020 PAKKA(KORTTI) = 1
1030 RETURN
```

Huomaa, että DIM-lauseke on ohjelman ensimmäisellä rivillä. Tämä johtuu siitä, että taulukko voidaan mitoittaa vain kerran eli sen ulottuvuutta ei voida asettaa uudelleen myöhemmin ohjelmassa.

Taulukon alkioiden arvoksi asetetaan 0 riveillä 20 - 40. Riviltä 1000 alkava aliohjelma valitsee kortin umpimähkään ja tarkistaa, ettei korttia ole jo aiemmin jaettu pakasta. Jos kortti on jo aiemmin jaettu, aliohjelma valitsee uuden kortin niin kauan kunnes se löytää pakasta aiemmin jakamattoman kortin. Seuraavaksi aliohjelma vaihtaa kyseisen taulukon alkion arvon ykköseksi. Tämän tarkoituksena on osoittaa, että kortti on nyt jaettu pakasta. Lopuksi aliohjelma palaa pääohjelmaan.

Taulukoiden käsittely ei suinkaan rajoitu yksiulotteisiin taulukoihin, vaan ulottuvuuksia voidaan lisätä tarpeen mukaan. Tämä saadaan aikaan lisäämällä muuttujan viitenumeroita. Esimerkiksi 10x10x10-taulukko (eli matriisi) saadaan aikaan seuraavalla käskyllä:

```
DIM TAULUKKO(10,10,10)
```

Tämä on kätevä tapa jakaa suureen ryhmään kuuluvat tiedot pienemmiksi alaryhmiksi. Korttipeliin liittyvässä esimerkissämme voisimme jakaa pakan neljään maahan, joihin kuhunkin kuuluu 13 korttia. Tällä tavalla jokaiseen korttiin päästäisiin erikseen käsiksi. Tämä saadaan aikaan käyttämällä käskymuotoa:

```
DIM PAKKA(4,13)
```

Jos nyt haluaisimme löytää vaikkapa ristinelosen, joka olisi voinut olla alkio 43 alkuperäisessä taulukossamme, saamme sen esille määrittelemällä yksinkertaisesti alkion (2,4) - olettaen, että ristit muodostavat toisen "rivin" uudessa taulukossamme.

Taulukoita ei suinkaan tarvitse käyttää pelkkien numerotietojen tallennukseen vaan niitä voidaan käyttää myös aivan yhtä hyvin merkkijonojen käsittelyyn. Tällaisesta sovellutuksesta voitaisiin mainita esimerkkinä vaikkapa matkustajien nimien kirjaaminen jonkun lentoyhtiön tietylle lennolle, tai teatterin istumapaikkojen varausjärjestelmä.

4.12 TIEDOT (DATA)

Tietoja voidaan syöttää ohjelmaan käyttämällä DATA-käskyä yhdessä READ-käskyn kanssa. Tarvittavat tiedot luetellaan ohjelmarivillä, jossa kukin yksittäinen osatieto erotetaan pilkulla. Koko luettelon eteen lisätään DATA-käsky. Tämän jälkeen tiedot saadaan käyttöön toinen toisensa perään READ-käskyllä.

Esimerkkiohjelma:

```
10 READ X,Y,Z
20 PRINT X;"+";Y;"+";Z;"=";X+Y+Z
30 DATA 1,3,5
```

DATA-lausekkeessa annettavat tiedot voivat olla numerojonoja tai kirjainjonoja tai niiden yhdistelmiä. Ei haittaa mitään, vaikka kaikki tiedot eivät mahtuisikaan yhdelle riville. Tällöin aloitetaan vain yksinkertaisesti uusi rivi ja laitetaan sen alkuun DATA-käsky. Tietokone käy READ-käskyn kohdatessaan koko ohjelman läpi ja etsii seuraavan osatiedon sen sijaintipaikasta riippumatta. Muista pitää huoli siitä, että kaikille READ-käskyille on riittävästi tietoja DATA-lausekkeissa, tai muuten tietokone tulostaa virheilmoituksen.

Tietojen peräkkäisessä järjestyksessä tapahtuva syöttö voidaan keskeyttää ainoastaan käyttämällä RESTORE-käskyä. Se asettaa dataosoittimen uudelleen ohjelman alkuun, jolloin samoja tietoja voidaan tarvittaessa käyttää useampaan kertaan. Seuraava ohjelma esittelee DATA-, READ- ja RESTORE-käskyjen käyttöä:

```
10 FOR C =1 TO 4
20 READ X$
30 PRINT X$;" ";
40 NEXT C
50 RESTORE
55 PRINT
60 GOTO 10
70 DATA HEI,KUINKA,SINULLA,HURISEE?
```

Ohjelma katkeaa painaessasi [ESC]-näppäintä kahdesti. DATA-lauseke voidaan haluttaessa sijoittaa mihin kohtaan hyvänsä ohjelmassa, vaikka esimerkkiohjelmassa DATA-lauseke onkin sijoitettu ohjelman loppuun.

DATA-käskyä voi käyttää muuhunkin kuin DATA-lausekkeesta luettavien tietojen tulostukseen PRINT-käskyllä. DATA-lausekkeessa olevia numeroarvoja voidaan lukea esimerkiksi SOUND-käskyyn seuraavasti:

```
10 FOR n=1 TO 30
20 READ s
30 SOUND 1,s,45,5
40 NEXT n
50 DATA 100,90,100,111,119,111,100,0
60 DATA 131,119,111,0,119,111,100,0
70 DATA 100,90,100,111,119,111,100,0
80 DATA 131,0,100,0,119,151
```

Muista säätää tietokoneen oikeassa päädyssä oleva äänenvoimakkuuden säädin sopivaan asentoon ennen kuin käynnistät ohjelman.

Päätämme tämän lyhyen BASIC-kieltä esittelevän jakson ohjelmaan, jonka avulla voit pelata venttiä CPC 464:ää (eli taloa) vastaan. (Pelin muita nimiä ovat mm. Vingt et un, kaksikymmentäyksi, Black Jack ja Pontoon, ja siitä on runsaasti eri versioita. Niinpä tämä venttiohjelma saattaakin poiketa siitä ventistä, jota ehkä itse olet tottunut palaamaan). Ohjelmassa esitellään lukuisia BASIC-kielen piirteitä, joiden ymmärtämistä helpottaa muuttujille valitut nimet. Voit täydentää ohjelmaa grafiikalla ja lisätä jännitystä äänitehosteilla.

Pelissä on tarkoituksena päästä mahdollisimman lähelle kahtakymmentäyhtä lisäämällä kädessäsi olevien korttien silmälukua. Talo puolestaan yrittää vastata peliisi yrittäen päästä vielä lähemmäksi kahtakymmentäyhtä. Kahtakymmentäyhtä ei saa ylittää, koska silloin peli menee metsään ja menetät panoksesi. Näppäile ensin rivit 1000 - 1020, ja aja tämä pieni aliohjelma RUN 1000-käskyllä. Näin saat käyttöösi ison Ä-kirjaimen painamalla [-näppäintä ja SHIFT-näppäintä. Käytä ohjelmaa näppäillessäsi AUTO-käskyä näppäiltyäsi ensin rivin 1, jolloin tietokone muodostaa rivinumerot automaattisesti.

```
1 REM VENTTI
2 GOSUB 1000
10 REM ALUSTUS
20 YC=2:CC=2
30 ACES=0
40 CACES=0
50 S=0
60 T=0
70 DIM MAA$(4)
80 MAA$(1)="RISTI"
90 MAA$(2)="HERTTA"
100 MAA$(3)="PATA"
110 MAA$(4)="RUUTU"
120 CLS
130 DIM PAKKA (52)
140 FOR X=L TO 52
150 PAKKA (X)=0
160 NEXT X
170 REM KAKSI KORTTIA JAETAAN MOLEMMILLE
 PELAAJILLE
180 LOCATE 3,3
190 PRINT "SINÄ";SPC(17)"TALO"
200 LOCATE 3,5
210 GOSUB 740
220 S=S+F
230 IF F=11 THEN ACES=ACES+1
240 LOCATE 3,6
250 GOSUB 740
260 S=S+F
270 IF F=11 THEN ACES=ACES+1
280 LOCATE 24,5
290 GOSUB 740
300 T=T+F
310 IF F=11 THEN CACES=CACES+1
320 LOCATE 24,6
330 GOSUB 740
340 T=T+F
350 IF F=11 THEN CACES=CACES+1
```

```
360 REM KIERTO-MAHDOLLISUUS (K) TAI MAHD
OLLISUUS PELATA OMILLAAN (O)
370 X$=INKEY$:IF X$<>"O" AND X$<>"K" THE
N 370
380 IF X$="O" THEN 560
390 LOCATE 3,YC+5
400 YC=YC+1
410 GOSUB 740
420 S=S+F
430 IF F=11 THEN ACES=ACES+1
440 REM PISTEIDEN JA ÄSSIEN TARKISTUS
450 IF S<22 THEN 370
460 IF ACES = 0 THEN 500
470 ACES = ACES-1
480 S=S-10
490 GOTO 450
500 LOCATE 12,19
510 PRINT "MENETIT PANOKSESI!"
520 PRINT:LOCATE 5,21:PRINT"HALUATKO PEL
ATA UUDESTAAN (K/E)?"
530 X$=INKEY$:IF X$<>"K" AND X$<>"E" THE
N 530
540 IF X$="K" THEN RUN
550 END
560 IF T>16 THEN 700
570 CC=CC+1
580 LOCATE 24,CC+4
590 GOSUB 740
600 T=T+F
610 IF F=11 THEN CACES=CACES-1
620 IF T<21 THEN 560
630 IF CACES = 0 THEN 670
640 CACES = CACES-1
650 T=T-10
660 GOTO 620
670 LOCATE 11,19
680 PRINT "VOITTO TULEE SINULLE"
690 GOTO 520
700 LOCATE 11,19
710 IF T<S THEN 680
720 PRINT "VOITTO MENEE TALOLLE"
730 GOTO 520
```

```
740 REM KORTIN JAKO
750 KORTTI=INT(RND(1)*52+1)
760 IF PAKKA(KORTTI)=1 THEN 750
770 PAKKA(KORTTI)=1
780 M$=MAA$(INT((KORTTI-1)/13)+1)
790 F=KORTTI-13*INT(KORTTI/13)
800 IF F=0 THEN F=13
810 IF F=1 OR F>10 THEN 850
820 PRINT M$;F;
830 IF F>10 THEN F=10
840 RETURN
850 IF F=11 THEN PRINT M$;"SOTAMIES";
860 IF F=12 THEN PRINT M$;"ROUVA";
870 IF F=13 THEN PRINT M$;"KUNINGAS";
880 IF F<>1 THEN 830
890 F=11
900 PRINT M$;"ÄSSÄ";
910 RETURN
1000 SYMBOL AFTER 91
1010 SYMBOL 123,102,24,60,102,126,102,102
1020 RETURN
```

Pelissä on mahdollisuus ns. kiertoon painamalla K-näppäintä sen jälkeen kun molemmille on jaettu kaksi korttia. Voit myös tyytyä jaettuihin kortteihin eli pelata omillasi painamalla O-näppäintä. (Huomaa, että tietokone hyväksyy vastaukseksi vain isot kirjaimet). Peli ei ole tarkoitettu suinkaan miksikään ehdottomaksi ja lopulliseksi vaan paremminkin erääksi ehdotukseksi venttipelin pelaamiseksi. Niinpä siihen voikin lisätä grafiikka- ja äänitehosteita, tai pelin kulun voi muuttaa omien venttisääntöjen mukaiseksi.

4.13 LOOGISET LAUSEKKEET

Tietokoneen ja laskimen välinen suurin ero on siinä, että tietokone pystyy käsittelemään loogisia operaatioita ehdollisten IF...THEN -lausekkeiden tapaisissa sovellutuksissa. Vertailuja tehtäessä loogiset operaattorit käsittelevät vertailtavia arvoja bittikuvioina ja vertailu tapahtuu yksittäisten bittien tasolla. Loogisia operaatioita on vaikeaa kuvata yksinkertaisesti, mutta asiat selvinnevät parhaiten tässäkin käytännön esimerkein sovittuamme ensin tiettyjen nimikkeiden käytöstä.

Looginen lauseke koostuu kahdesta puolikkaasta, joita kutsutaan argumenteiksi (ne ovat usein muuttujia). Niiden välissä on looginen operaattori, joten loogisen lausekkeen muoto on seuraava:

<argumentti >[< looginen operaattori><argumentti>]

jossa:

<argumentti > on: NOT <argumentti > (EI <argumentti>)
tai: <numerolauseke >
tai: <vertailulauseke >
tai: (<looginen lauseke>)

Tietokone muuttaa molemmat loogisella operaattorilla vertailtavat argumentit aina kokonaislukumuotoon ja antaa virheilmoituksen **Error 6** jos argumentti ei mahdu tietokoneen kokonaislukualueelle.

Loogiset operaattorit ovat (arvojärjestyksessä ja siinä järjestyksessä, jossa ne vaikuttavat kuhunkin bittiin):

AND	Tulos on 0 paitsi jos molempien argumenttibittien arvo on 1
OR	Tulos on 1 paitsi jos molempien argumenttibittien arvo on 0
XOR	Tulos on 1 paitsi jos molempien argumenttibittien arvo on sama

(AND = JA, OR = TAI, XOR = POISSULKEVA TAI).

AND on kaikkein yleisimmin käytetty looginen operaattori, eikä sen merkitys ole suinkaan sama kuin "laske yhteen".

PRINT 10 AND 10 antaa tulokseksi 10.

PRINT 10 AND 12 antaa tulokseksi 8.

PRINT 10 AND 1000 antaa tulokseksi jälleen 8.

Näin käy siksi, että luvut 10 ja 1000 on muunnettu binäärimuotoon:

```
      1010
1111101000
```

AND-operaatio tarkistaa molemmista vertailtavista luvuista (argumenteista) jokaisen toisiaan vastaavan bitin yksi kerrallaan ja antaa tuloksen 1 vain sellaisessa kohdassa, jossa bitin arvo on 1 sekä ylemmässä että alemmassa rivissä:

```
0000001000
```

Tämä luku on juuri 8, kun se muunnetaan takaisin desimaalimuotoon. Tämä siis merkitsee sitä, että loogista operaattoria AND käytetään tunnistamaan sellainen tilanne, jossa kaksi ehtoa ovat voimassa samanaikaisesti. Seuraava ohjelma kuvaa AND-operaattorin käyttöä:

```
10 CLS:INPUT "Monesko paiva nyt on";paiv
a
20 INPUT "Monesko kuukausi nyt on";kuuka
usi
30 IF paiva=25 AND kuukausi=12 GOTO 50
40 GOTO 10
50 PRINT "Hauskaa Joulua!"
```

Myös OR-operaattori vertailee argumenttien bittejä antaen tulokseksi 1 paitsi silloin, kun molemmat argumenttibitit ovat 0, jolloin tulokseksi saadaan 0. Seuraavassa esimerkissä on käytetty samoja lukuja kuin AND-operaattoria koskevassa esimerkissäkin:

```
PRINT 10 OR 1000        antaa tulokseksi 1002
```

Bittitasolla tilanne näyttää seuraavalta:

```
      1010
1111101000              antaa tulokseksi 1111100010
```

Ohjelmaesimerkki:

```
10 CLS
20 INPUT "Monesko kuukausi nyt on";kuuka
usi
30 IF kuukausi=12 OR kuukausi=1 OR kuuka
usi=2 GOTO 50
40 GOTO 10
50 PRINT "Nyt on varmaankin talvi."
```

```
10 CLS
20 INPUT "Monesko kuukausi nyt on";kuuka
usi
30 IF NOT(kuukausi=6 OR kuukausi=7 OR ku
ukausi=8) GOTO 50
40 GOTO 10
50 PRINT "Nyt ei ole ainakaan kesa!"
```

Loogisia operaattoreita on mahdollista sijoittaa peräkkäin samalle riville niin monta kuin tarvitaan (rivin maksimipituuden puitteissa), jolloin seula saadaan tiheämmäksi ja vain kaikki vertailun ehdot täyttävä tulos aiheuttaa toimenpiteitä. Esimerkki:

```
10 CLS:INPUT "Monesko paiva nyt on";paiv
a
20 INPUT "Monesko kuukausi nyt on";kuuka
usi
30 IF NOT(kuukausi=12 OR kuukausi=1) AND
 paiva=29 GOTO 50
40 GOTO 10
50 PRINT "Nyt ei ole joulukuu eika tammi
kuu, mutta nyt saattaisi olla karkausvuo
si."
```

Vertailulausekkeen tulos on joko -1 tai 0. Tulosta -1 vastaa bittitasolla se, että kaikki kokonaisluvun bitit = 1. Vastaavasti tulosta 0 vastaa se, että kaikki kokonaisluvun bitit = 0. Kahden tällaisen argumentin loogisen operaation tulos antaa tulokseksi joko -1 (tosi) tai 0 (epätosi).

Voit tarkistaa tämän lisäämällä edellä olevaan ohjelmaan seuraavat rivit:

```
60 PRINT NOT(kuukausi=12 OR kuukausi=1)
70 PRINT (kuukausi=12 OR kuukausi=1)
```

Kun nyt ajat ohjelman ja vastaat kysymyksiin vaikkapa siten, että päivä = 29 ja kuukausi = 5, saat ruutuun rivillä 50 olevan vastauksen. Lisäksi sen alle ilmestyvät rivien 60 ja 70 mukaiset loogisten lausekkeiden todelliset arvot.

Lopuksi käsittelemme loogista operaattoria XOR (EXCLUSIVE OR = POISSULKEVA TAI), joka antaa tulokseksi tosi niin kauan kun molemmat argumentit ovat erilaiset. Loogisten operaattoreiden ominaisuudet on kerätty alla olevaan taulukkoon, jota kutsutaan totuustaulukoksi. Siitä näkee kätevästi loogisten operaattorieden vaikutuksen eri argumenttien bittiarvoille.

Argumentti A	1	0	1	0
Argumentti B	0	1	1	0
Tulos AND-operaattorilla	0	0	1	0
Tulos OR-operaattorilla	1	1	1	0
Tulos XOR-operaattorilla	1	1	0	0

LUKU 5: GRAFIIKAN ALKEET

Tässä luvussa käsitellään seuraavia CPC 464:n grafiikkaan liittyviä aiheita:

- Skandinaaviset merkit
- Näyttömuodot (MODE) ja kuva-alkiot (PIXEL)
- Värit
- INK-, PAPER- ja PEN-käskyt
- Viivojen piirtäminen
- Ikkunat (WINDOW)

5.1 CPC 464:LLE TYYPILLISIÄ OMINAISUUKSIA

Olemme käsitelleet tähän mennessä AMSTRADin BASIC-ohjelmia ja sovellutuksia lähinnä useimpiin muihinkin mikrotietokoneisiin soveltuvien käskyjen osalta. Suurin osa puhtaasti aritmeettisista operaatioista toimii sellaisenaan tai pienin muutoksin lähes kaikissa BASIC-kielen murteissa. Grafiikkatoimintoja (ja tekstikohdistimen sijaintia) ohjaavat BASIC-käskyt ovat kuitenkin enemmän laitekohtaisia. Tämä pätee myös CPC 464:n kohdalla. Niinpä grafiikkatoimintojen ohjauskäskyihin onkin syytä perehtyä huolellisesti, jotta saisit mikrosi näytönohjaustoiminnoista kaiken irti.

BASICin avainsanat on esitetty tässäkin luvussa muusta tekstistä erottuvalla tavalla (suurilla kirjaimilla), ja luvusta 8 löytyy lisää esimerkkejä avainsanojen käytöstä. Sieltä löytyy myös lyhyt kuvaus kunkin avainsanan merkityksesta.

5.1.1 Skandinaaviset merkit

Skandinaaviset merkit å, ä, ö, Å, Ä ja Ö on helppo ohjelmoida CPC 464:llä. Ohjelma kannattaa tallentaa jonkun kasetin alkuun, josta se aina tarvittaessa kätevästi löytyy. Ohjelma näyttää seuraavalta:

```
5 REM skandinaaviset merkit
10 SYMBOL AFTER 91
20 SYMBOL 92,48,0,120,12,124,204,118
30 SYMBOL 91,204,0,120,12,124,204,118
40 SYMBOL 93,102,0,60,102,102,102,60
50 SYMBOL 96,24,0,60,102,126,102,102
60 SYMBOL 123,102,24,60,102,126,102,102
70 SYMBOL 125,102,24,60,102,102,60,24
```

Rivillä 10 määritellään se näppäin (91), jonka jälkeen merkit muutetaan oletusarvoista. Riveillä 20 - 70 määritellään itse merkit. Ohjelmarivin ensimmäinen numero määrittelee sen näppäimen, jonka kohdalta kyseinen merkki löytyy. Loput 7 numeroa määrittelevät itse merkin. Näin määriteltyinä pienet kirjaimet å, ä ja ö löytyvät näppäinten \, [ja], sekä isot kirjaimet Å, Ä ja Ö löytyvät vastaavien näppäinten kohdalta painettaessa SHIFT-näppäintä.

Joissakin tapauksissa voi osoittautua tarpeelliseksi ohjelmoida å, ä, ö, Å, Ä ja Ö näppäinten }, {, |,], [ja \. Tällainen tilanne voi tulla vastaan, kun CPC 464 liitetään "ulkomaailmaan" eli joihinkin kirjoittimiin tms. Lisätietoja löytyy mm. luvusta 8 ja liitteestä III.

Tästä eteenpäin olevissa ohjelmalistauksissa on käytetty skandinaavisia merkkejä. Jos haluat käyttää niitä myös omissa ohjelmissasi, sinun on ensin ajettava edellä oleva ohjelma RUN-käskyllä. Ohjelma kannattaa siis tallentaa jonkun kasetin alkuun, josta se löytyy tarvittaessa.

Merkit pyyhkiytyvät pois muistista ja tietokone palaa alkutilaan (oletusarvoihin), jos nollaat tietokoneen painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä.

5.1.2 Värien valinta

Seuraavissa kappaleissa myös mustaa väriä käsitellään muiden värien tapaan puhuttaessa väreihin liittyvistä käskyistä.

Reuna-alueen väriksi voidaan valita mikä hyvänsä väri tai väripari BORDER-käskyllä näyttömuodosta riippumatta. Reuna-alueen väri ei vaihdu, vaikka näyttömuotoa vaihdetaan MODE-käskyllä. Reuna-alueen väri voi olla kiinteä, tai se saadaan vilkkumaan halutun väriparin väreissä.

Näytössä voi olla samanaikaisesti useampia musteen (INK) värejä. Samanaikaisesti esitettävien värien määrä riippuu valitusta näyttömuodosta. Kukin musteen väri voi näkyä kiinteästi, tai se saadaan vilkkumaan halutun väriparin väreissä. Käytettävissä olevien musteen värien määrä riippuu siis kullakin hetkellä käytössä olevasta näyttömuodosta. Kirjoituksen pohjana olevan "paperin" (PAPER), kirjoitukseen käytettävän "kynän" (PEN) ja grafiikkaan käytettävän kynän (PEN) väri voidaan sitten valita käytettävissä olevista väreistä. Väri valitaan musteen (INK) väreistä.

5.1.3 Läpinäkyvä näyttötapa sekä PEN-, INK- ja PAPER-käskyjen välinen suhde

Kun merkkejä tulostetaan ruutuun, käytetään kahta INK-käskyllä määriteltävää väriä. Toisella INK-käskyllä määritellään kynän (PEN) väri ja toisella paperin (PAPER) väri. Tämä ei koske tilannetta, jossa näyttöön on määritelty kahden värin välillä tapahtuva vilkutus.

> HUOM!
> PAPER-käskyyn liittyvä numero määrittelee kyseisen musteen (INK) numeron EIKÄ värin numeroa, jotka on lueteltu liitteessä IV. Vastaavasti PEN-käskyyn liittyvä numero määrittelee kyseiselle kynälle (PEN) kuuluvan musteen (INK) numeron EIKÄ värin numeroa liitteessä IV kuvatulla tavalla.

Paperin (PAPER) oletusarvo on 0, ellei muuta ole määritelty. Vastaavasti kynän (PEN) oletusarvo on 1. Jos haluat vaihtaa paperin nro 0 musteen (INK) vihreäksi, joka on väri nro 9, näppäile:

```
INK 0,9
```

Vastaavasti kynän nro 1 muste vaihdetaan mustaksi (väriksi nro 0) näppäilemällä:

```
INK 1,0
```

Koko näyttö pimenee, jos asetat paperin samanväriseksi kynän kanssa INK-käskyllä INK 0,0.

Ruutuun "kirjoittuva" teksti voidaan saada läpinäkyväksi, jollainen kohdistimen virkaa hoitava neliö aina on. Tämä tapahtuu käyttämällä tiettyjä ohjausmerkkejä, joilla BASICin grafiikkakäskyihin saadaan uusia ulottuvuuksia. Läpinäkyvää näyttötapaa käyttäessäsi voit joko kokonaan unohtaa paperin värin ja kirjoittaa grafiikkakuvoiden päälle tai kirjoittaa taustan päälle aivan kokonaan. Seuraava lyhyt ohjelma esittelee läpinäkyvän näyttötavan käyttöä:

(Jos olet ohjelmoinut skandinaaviset merkit luvussa 5.1.1 kuvatulla tavalla, pyyhkiytyvät ne pois muistista, jos nollaat tietokoneen painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä).

```
new
10 MODE 1
20 INK 2,19
30 DRAW 200,200,2
40 LOCATE 1,21
50 PRINT "1 NORMAALI"
60 PRINT CHR$(22)+CHR$(1)
70 ORIGIN 0,0
80 DRAW 500,200,2
90 LOCATE 12,18
100 PRINT"2 LÄPINÄKYVÄ"
110 PRINT CHR$(22)+CHR$(0)
120 LOCATE 22,15
130 PRINT"3 TAAS NORMAALI"
```

Ohjelma piirtää ensimmäisen viivan rivillä 30 olevalla DRAW-käskyllä ennen läpinäkyvän näyttötavan valintaa. Toinen viiva on sensijaan piirretty rivillä 80 olevalla DRAW-käskyllä sen jälkeen kun läpinäkyvä näyttötapa on valittu rivillä 60 olevalla käsky-yhdistelmällä CHR$(22)+CHR$(1). Huomaa, kuinka väri on vaihtunut päällekkäisissä kohdissa ja kuinka viiva on piirtynyt täydellisesti merkkisolun (8 x 8 pistettä) täyttävän musteen yli läpinäkyvässä näyttötavassa.

Vaihda läpinäkyvän näyttötavan aloitusrivi (rivi 60) ja lopetusrivi (rivi 110) toiseen paikkaan ohjelmassa niin näet, miten tämä vaikuttaa näyttöön. Kaikki muutkin ohjausmerkit on lueteltu liitteessä III, ja niitä käsitellään tarkemmin luvussa 9.

5.2 NÄYTTÖMUODOT (MODE)

CPC 464:ssä on kolme näyttömuotoa, joissa tekstiä ja grafiikkaa tulostetaan kuvaruutuun:

a) Normaali näyttömuoto (tietokoneen valitsema oletusarvo) eli MODE 1:

- 40 saraketta x 25 riviä; 4 musteen väriä (INK) tekstille
- 320 x 200 kuva-alkiota (pixel), kukin erikseen osoitettavissa, käytettävissä 4 eri väriä

b) Monivärinen näyttömuoto eli MODE 0:

- 20 saraketta x 25 riviä; 16 musteen väriä (INK) tekstille
- 160 x 200 kuva-alkiota, kukin erikseen osoitettavissa, käytettävissä 16 eri väriä

c) Suuren erottelukyvyn näyttömuoto eli MODE 2:

- 80 saraketta x 25 riviä; 2 musteen väriä (INK) tekstille
- 640 x 200 kuva-alkiota, kukin erikseen osoitettavissa, käytettävissä 2 eri väriä

Kuten näet, ero on näytössä näkyvien vaakasuunnan kuva-alkioiden lukumäärässä. Tätä ei pidä sekoittaa TV-kuvaputken pinnalla näkyviin pieniin vaakasuoriin juoviin, koska se on kuvaputken rakenteesta johtuva ominaisuus.

Nimitämme näitä erilaisia näyttötapoja näyttömuodoiksi (MODE), ja näytössä voi olla käytössä vain yksi näyttömuoto kerrallaan käytettäessä BASICia. Näyttömuodon vaihtaminen tyhjentää kuvaruudun aivan kokonaan. Tämä koskee myös kuvaruudussa näkyviä tekstejä ja grafiikkaikkunoita. Vaikutus on sama kuin CLS- ja CLG-käskyillä. Näyttömuodon vaihtaminen ei vaikuta ohjelmamuistin sisältöön.

Näyttömuotoa voidaan vaihtaa BASIC-ohjelmaan sijoitetulla käskyllä tai näppäimistöltä annettavalla suoralla käskyllä.

5.2.1 Monivärinen grafiikkamuoto MODE 0

Tässä näyttömuodossa voidaan käyttää samanaikaisesti kuuttatoista 27:stä käytettävissä olevasta väristä. Kunkin yksittäisen kuva-alkion väri on erikseen ohjelmoitavissa. Näyttö koostuu 160 kuva-alkiosta vaakasuunnassa ja 200 kuva-alkiosta pystysuunnassa. Kuva-alkioruudukko on esitetty liitteessä VI.

Näyttömuodossa 0 on 25 riviä, ja kullakin niistä on 20 merkkiä.

5.2.2 Oletusarvona valittava "tavallinen" näyttömuoto MODE 1

Näyttömuodoksi tulee aina MODE 1, kun virta kytketään CPC 464:ään. Näytössä voi olla samanaikaisesti neljä 27 väristä, ja voit vaihtaa näyttöön nopeasti minkä hyvänsä 27 väristä ohjelmakäskyllä. Kuvaruudussa on 320 kuva-alkiota vaakasuunnassa ja 200 pystysuunnassa. Kuva-alkioruudukko on esitetty liitteessä VI.

Näyttömuodossa 1 on 25 riviä, joilla kullakin on 40 merkkiä.

5.2.3 Suuren erottelykyvyn näyttömuoto MODE 2

Tässä näyttömuodossa voit käyttää samanaikaisesti näytössä kahta eri väriä. Tätä näyttömuotoa käytetään etupäässä silloin, kun tarvitaan 80 tekstimerkkiä riville. Tällöin esimerkiksi ohjelman kirjoittaminen on huomattavasti helpompaa, koska ohjelmalistauksesta näkyy kerralla huomattavasti enemmän muihin näyttömuotoihin verrattuna. Kuvaruudussa on 640 kuva-alkiota vaakasuunnassa ja 200 pystysuunnassa.

Näyttömuodossa 2 on 25 riviä, joilla kullakin on 80 merkkiä.

5.2.4 Esittelyohjelmia

Tyhjennä näyttö ja ohjelmamuisti CLS- ja NEW-käskyillä ja näppäile sitten seuraava ohjelma:

```
5 REM GRAFIIKAN ESITTELYOHJELMA
10 MODE 1
15 INK 2,0
16 INK 3,6: REM VÄRIN VALINTA
17 BORDER 1:REM TUMMANSININEN
20 CLG:REM NÄYTÖN TYHJENNYS
30 b%=RND*5+1 :REM ASETTAA SATUNNAISMUUT
TUJAT
40 c%=RND*5+1
50 ORIGIN 320,200 :REM GRAFIIKKAKOHDISTI
MEN ALKUPISTEEN ASETUS
60 FOR A = 0 TO 1000 STEP PI/30
70 x%=100*COS(a)
80 MOVE x%,x% :REM GRAFIIKKAKOHDISTIMEN
SIIRTO
90 DRAW 200*COS(a/b%),200*SIN(a/c%),3 :
REM VIIVAN PIIRTO
91 IF INKEY$<>"" THEN 20
100 NEXT :REM TAKAISIN RIVILLE 60 ELLEI
KESKEYTETÄ RIVILLÄ 91
110 GOTO 20
```

Aja ohjelma RUN-käskyllä. Saat uuden kuvion painamalla mitä hyvänsä näppäimistön näppäintä. Ohjelma esittelee useitakin CPC 464:n tärkeitä ominaisuuksia: CPC 464 tulostaa ruutuun hyvin pehmeästi ja tasaisesti, ja ohjelmistoon kuuluu sellaisia käskyjä, joilla saadaan aikaan varsin hienoja tehosteita hyvin vähällä vaivalla. REM-lausekkeita käytetään selitysten ja huomautusten tekemiseen. Ohjelma toimii ilman REM-lausekkeitakin, mutta ne auttavat ymmärtämään ohjelman tekijän ajatuksia.

Ohjelman rivinumeroista näkee, että siihen on lisätty jälkeenpäin käskyrivejä. Olisimme toki voineet "siistiä" sen RENUM-käskyllä, mutta näin näet paremmin, miten ohjelmaa on kehitetty alkuperäisestä ideasta.

Jos haluat tallentaa ohjelman kasetille, näppäile esim. seuraava käsky:

```
SAVE "GRAFIIKKAA050585"
```

(Muistathan, että tiedoston nimen pituus voi olla enintään 16 merkkiä välilyönnit mukaan luettuna. Ohjelma toki tallentuu myös antaessasi pitemmän nimen, mutta tietokone huomioi vain nimen 16 ensimmäistä merkkiä).

Seuraavaksi on grafiikan esittelyohjelma, joka piirtää erivärisiä interferenssikuvioita:

```
NEW
10 a$=INKEY$: REM UUSI KUVIO KÄYNNISTYY
PAINAESSASI JOTAIN NÄPPÄINTÄ
20 IF a$="" THEN 10
30 CLS
40 m=INT(RND*3):REM SATUNNAISLUVUN VALIN
TA
50 IF m>2 THEN 40:REM UUSI YRITYS JOS LU
KU ON SUUREMPI KUIN 2
60 MODE m
70 i1=RND*26:REM SATUNNAISEN VÄRIN (INK)
 VALINTA
80 i2=RND*26
90 IF ABS(i1-i2)<5 THEN 70
100 INK 0,i1:INK 1,i2
110 s=RND*5+3
120 ORIGIN 320,-100
130 FOR x= -1000 TO 0 STEP s
140 MOVE 0,0
150 DRAW x,300:DRAW 0,600
160 MOVE 0,0
170 DRAW -x,300: DRAW 0,600
180 a$=INKEY$
190 IF a$<>"" THEN 30:REM SILMUKKA KESKE
YTYY PAINETTAESSA JOTAIN NÄPPÄINTÄ
200 NEXT x
210 GOTO 10
```

(Ohjelma käynnistyy painettaessa mitä hyvänsä näppäintä RUN-käskyn jälkeen).

Tämä ja edellinen ohjelma esittelevät eräitä yksinkertaisia matemaattisia käsitteitä varsin värikkäästi ja visuaalisesti. Molemmat ohjelmat perustuvat sattumanvaraisesti muodostetun "siemen"-luvun perusteella laskettuun summaan. Tällä varmistetaan, että jokainen kuvio on hiukan erilainen. Ohjelma näyttää tulokset sattumanvaraisina viivoina.

CPC 464-tietokoneesi on myös erinomainen sähköinen grafiikkataulu, ja näin tietokoneesi piirtää erään tunnetuimmista grafiikkakuvioista eli sinikäyrän:

```
10 REM PIIRRETÄÄN SINIKÄYRÄ
20 MODE 2
30 INK 1,21
40 INK 0,0
50 CLS
60 DEG
70 ORIGIN 0,200
80 FOR n=0 TO 720
90 y=SIN(n)
100 PLOT n*640/720,198*y,1
110 NEXT
```

Rivillä 100 oleva PLOT-lauseke on se ohjelman osa, joka piirtää viivan. Lauseke tuottaa yhden pisteen (kuva-alkion) kuvaruutuun laskettuaan ensin pisteen arvon riveillä 80 - 110 olevalla FOR NEXT-silmukalla, ja tulostaa vastauksen ruutuun.

CPC 464:ssä on useita yksinkertaisia ja tehokkaita käskyjä. Niinpä voitkin lisätä erään tehosteen edellä olevaan ohjelmaan lisäämällä seuraavan rivin:

```
15 BORDER 6,9
```

Kun nyt ajat ohjelman RUN-käskyllä, alkaa reuna-alue (BORDER) vilkkua vuorotellen värien nro 6 ja nro 9 välillä. Vilkkumisnopeus määräytyy tietokoneen valitseman oletusarvon mukaan. Ohjelma saadaan toistumaan silmukassa yhä uudelleen ja uudelleen, jos lisäät siihen seuraavan rivin:

```
120 GOTO 50
```

Ohjelma keskeytyy painaessasi nyt [ESC]-näppäintä kerran ja katkeaa painaessasi näppäintä kahdesti. Reuna-alueen vilkkuminen ei pysähdy vaikka pysäyttäisitkin ohjelman. Tämä johtuu siitä, että reuna-alueen toimintaa ohjataan muusta ohjelmasta riippumatta. Voit lopettaa reuna-alueen vilkkumisen ja asettaa väriksi kirkkaansinisen painamalla [ESC]-näppäintä kahdesti ja muuttamalla rivin 15 seuraavasti:

```
15 BORDER 2
```

Vilkkuminen lakkaa, kun nyt ajat ohjelman RUN-käskyllä.

Jos haluat vaihtaa käyrän ja taustan väriä, sinun on vaihdettava riveillä 30 ja 40 olevien INK-käskyjen väriä. Ohjelman listauksen pitäisi nyt näyttää seuraavalta:

```
10 REM PIIRRETÄÄN SINIKÄYRÄ
15 BORDER 2
20 MODE 2
30 INK 1,2
40 INK 0,20
50 CLS
60 DEG
70 ORIGIN 0,200
80 FOR n=0 TO 720
90 y=SIN(n)
100 PLOT n*640/720,198*y,1
110 NEXT
120 GOTO 50
```

Rivillä 100 olevan PLOT-lausekkeen lopussa oleva numero 1 ilmoittaa tietokoneelle, että sen on piirrettävä käyrä rivillä 30 olevalla INK 1-käskyllä määritellyllä värillä. Katso luvussa 8 olevasta avainsanojen luettelosta PLOT-lausekkeen kuvausta, niin näet, kuinka lausekkeen eri osat vaikuttavat.

Jos katsot tarkkaan kuvaruutuun piirtyvää käyrää niin näet, ettei se ole yhtenäinen viiva vaan se muodostuu pienistä irrallisista osasista. Nämä pienet osaset ovat juuri kuva-alkioita, joista oli jo aiemmin puhetta.

5.2.5 Grafiikkakohdistin ja viivojen piirtäminen

Olet nyt kokeillut eräitä tapoja, joilla ohjelmat saadaan tulostamaan kuvaruutuun erilaisia grafiikkakuvioita. Ohjelmissa on käsitelty joukko ohjelmakäskyjä ja käsitteitä, joilla erilaisia tehosteita saadaan aikaan. Viivoja kuvaruutuun piirrettäessä kannattaa eräät seikat painaa visusti mieleen, etteivät ne aiheuttaisi turhaan sekaannusta.

Ensimmäinen huomion arvoinen seikka on ohjelmamuistin tämänhetkinen tila. Tietokone muistaa tämänhetkiset väriasetukset myös sen jälkeen, kun ohjelmamuisti on nollattu NEW-käskyllä. Tietokone voidaan nollata aivan alkutilaansa painamalla [CTRL]-, [SHIFT]- ja [ESC]-näppäimiä tässä järjestyksessä. Ennen tätä on kuitenkin ohjelmamuistissa oleva ohjelma tallennettava kasetille SAVE-käskyllä, mikäli ohjelmaa halutaan käyttää myöhemmin.

Voit kokeilla tätä näppäilemällä NEW:CLS sen jälkeen kun olet katkaissut edellisen ohjelman suorituksen [ESC]-näppäimellä. Näppäile nyt:

```
DRAW 100, 100
```

DRAW-käsky piirtää suoran viivan <u>grafiikkakohdistimen</u> viimeisimmästä paikasta X- ja Y-koordinaateilla määriteltyyn pisteeseen (100, 100). Grafiikkakohdistin on näkymätön, ja sen tehtävänä on osoittaa se kohta, jossa seuraava grafiikkatoiminto tapahtuu.

Saat selville grafiikkakohdistimen sijainnin käyttämällä XPOS- ja YPOS-toimintoja seuraavasti:

```
PRINT XPOS
```

Vastaus on 100. (Saat saman vastauksen myös kysymykseesi PRINT YPOS, koska X:n ja Y:n arvo on 100 kyseisessä pisteessä).

Jos teksti tulostuu aivan kuvaruudun alaosaan aiheuttaen näytön siirtymisen ylöspäin ("vierityksen" ylöspäin), siirtyy myös grafiikkanäyttö ylöspäin, mutta grafiikkakohdistin jää entiseen paikkaansa. Voit kokeilla tätä pitämällä kohdistimen alaspäinsiirtonäppäintä [↓], kunnes kuvaruutu tyhjenee tekstien siirryttyä tarpeeksi ylös. Tiedustele nyt uudelleen X- ja Y-koordinaattien arvoja XPOS- ja YPOS-avainsanoilla. Ja kuten huomaat, on grafiikkakohdistin edelleen aikaisemmassa kohdassa.

Voit lisätä DRAW-käskylausekkeen perään uuden argumentin, jolla voi määritellä piirrettävän viivan värin (katso edellisen sivun ohjelman jälkeen olevaa PLOT-käskyn kuvausta, koska toiminta on aivan samanlainen).

Ensiksi sinun on kuitenkin pitänyt määritellä musteen väri INK-käskyllä. Muistathan, että voit käyttää vain tiettyjä musteen värejä kulloinkin käyttämässäsi näyttömuodossa (MODE). Kokeile tätä näppäilemällä:

```
10 MODE 1
20 INK 0,10
30 ORIGIN 0,0
40 INK 1,26
50 INK 2,0
60 DRAW 320,400,1
70 DRAW 640,0,2
```

Seuraavassa esimerkkiohjelmassa on käytetty kaikkia tähän mennessä mainittuja asioita, ja siihen on otettu mukaan muutamia uusiakin ominaisuuksia. Ensimmäisellä ohjelmarivillä (10) asetetaan väri- ja INK-arvot sellaisiksi, että halutut tulokset saavutetaan siitä riippumatta, mitkä väriarvot CPC 464:n muistissa on ennen ohjelman käynnistämistä:

```
10 INK 0,0:INK 1,26:INK 2,6:INK 3,18: BO
RDER 0
20 REM ohjelma piirtää kuvioita
30 MODE 1:DEG
40 PRINT"3-, 4- vai 6-sivuinen kuvio?";
50 LINE INPUT p$
60 IF p$="3" THEN sa=120:GOTO 100
70 IF p$="4" THEN sa=135:GOTO 100
80 IF p$="6" THEN sa=150:GOTO 100
90 CLS:GOTO 40
100 PRINT:PRINT "Odota hetki";
105 IF p$="3" THEN ORIGIN 0,-50,0,640,0,
400 ELSE ORIGIN 0,0,0,640,0,400
110 DIM cx(5),cy(5),r(5),lc(5)
120 DIM np(5)
130 DIM px%(5,81),py%(5,81)
140 st=1
150 cx(1)=320:cy(1)=200:r(1)=80
160 FOR st=1 TO 4
170 r(st+1)=r(st)/2
180 NEXT st
190 FOR st=1 TO 5
200  lc(st)=0:np(st)=0
210  np(st)=np(st)+1
220  px%(st,np(st))=r(st)*SIN(lc(st))
230  py%(st,np(st))=r(st)*COS(lc(st))
240  lc(st)=lc(st)+360/r(st)
245  IF(lc(st) MOD 60)=0 THEN PRINT".";
250  IF lc(st) < 360 THEN 210
252  px%(st,np(st)+1)=px%(st,1)
254  py%(st,np(st)+1)=py%(st,1)
260 NEXT st
265 CLS:INK 1,2
270 st=1
280 GOSUB 340
290 LOCATE 1,1
300 EVERY 25,1 GOSUB 510
310 EVERY 15,2 GOSUB 550
320 EVERY 5,3 GOSUB 590
330 GOTO 330
340 REM piirretään ympyrä ja 3,4 tai 6
muuta ympyrää sen ympärille
350 cx%=cx(st):cy%=cy(st):lc(st)=0
360 FOR x%=1 TO np(st)
370 MOVE cx%,cy%
```

```
380 DRAW cx%+px%(st,x%),cy%+py%(st,x%),
1+(st MOD 3)
390 DRAW cx%+px%(st,x%+1),cy%+py%(st,x%
+1),1+(st MOD 3)
400 NEXT x%
410 IF st=5 THEN RETURN
420 lc(st)=0
430 cx(st+1)=cx(st)+1.5*r(st)*SIN(sa+lc(
st))
440 cy(st+1)=cy(st)+1.5*r(st)*COS(sa+lc(
st))
450 st=st+1
460 GOSUB 340
470 st=st-1
480 lc(st)=lc(st)+2*sa
490 IF (lc(st) MOD 360)<>0 THEN 430
500 RETURN
510 ik(1)=1+RND*25
520 IF ik(1)=ik(2) OR ik(1)=ik(3) THEN 5
10
530 INK 1,ik(1)
540 RETURN
550 ik(2)=1+RND*25
560 IF ik(2)=ik(1) OR ik(2)=ik(3) THEN 5
50
570 INK 2,ik(2)
580 RETURN
590 ik(3)=1+RND*25
600 IF ik(3)=ik(1) OR ik(3)=ik(2) THEN 5
90
610 INK 3,ik(3)
620 RETURN
```

Kun käynnistät ohjelman RUN-käskyllä, ilmestyy kuvaruutuun kysymys (riviltä 40). Vastaa 3, niin näet tulokset nopeimmin. Tällöin ohjelma tulostaa ruutuun viestin **Odota hetki** . Samoin viestin perään ilmestyy piste muutaman sekunnin välein (riviltä 245) sen merkiksi, että tietokone suorittaa parhaillaan laskutoimituksia. Samalla näet, että ohjelma on toiminnassa.

Riveiltä 300 - 320 kutsuttavat aliohjelmat saavat aikaan eri värien vilkkumisen. Vilkkumisnopeus määritellään EVERY-käskyllä. Jos haluat hidastaa vilkkumisnopeutta, korjaa rivit 300 - 320 esim. seuraavaan muotoon:

```
300 EVERY 250,1 GOSUB 510
310 EVERY 150,2 GOSUB 550
320 EVERY 50,3 GOSUB 590
```

Jos haluat tietää enemmän EVERY-käskystä, katso lukua 8. EVERY-käsky on nimittäin eräs AMSTRADin BASICin käyttökelpoisimmista piirteistä. Eräs EVERY-käskyn mielenkiintoinen piirre on sen tapa kasata tehtäviä sillä aikaa, kun ohjelma keskeytetään painamalla ESC -näppäintä <u>kerran</u>.

Keskeytä ohjelman suoritus muutamaksi sekunniksi painamalla ESC -näppäintä, ja käynnistä ohjelma uudelleen painamalla mitä muuta näppäintä hyvänsä. Näyttö vilkkuu nyt vinhasti sen ajan, jonka EVERY-käskyltä menee "ruuhkautuneiden tehtävien" purkamiseen. Jonoon mahtuu vain rajallinen määrä tehtäviä, joten jonkin ajan kuluttua käy niin, että uusi EVERY-käsky jää huomioimatta, kunnes jonossa on jälleen tilaa ruuhkautuneiden tehtävien tultua puretuksi.

5.3 IKKUNAT (WINDOW)

Käyttäjä voi valita jopa kahdeksan teksti-ikkunaa, joihin merkkejä voidaan tulostaa. Samoin käytettävissä on grafiikkaikkuna, johon voidaan tulostaa grafiikkakuvioita. Ikkunat palautuvat oletusarvoihinsa (alkutilaan) asetettaessa kuvaruutu uuteen näyttömuotoon (MODE). Asiaa käsitellään tarkemmin luvussa 8, jossa on eri avainsanojen kuvaukset.

HUOM!
Jos teksti-ikkuna on yhtä suuri kuin koko kuvaruutu (oletusarvo), niin kuvan nopea vieritys saadaan aikaan laitteiston omilla toiminnoilla. Jos taas teksti-ikkuna on pienempi kuin käytettävissä oleva kuvaruutu, niin vieritys saadaan aikaan ohjelmallisesti, mikä on hitaampaa kuin koko kuvan vieritys.

WINDOW-käskyllä määritellään halutun tulostusreitin (ikkunan) vasemmanpuoleisin, oikeanpuoleisin, ylin ja alin merkkisolu. Ikkunat voivat olla osittain tai kokonaan toistensa päällä. Ikkunoiden avulla saadaan nopeasti tehdyksi värillisiä laatikoita. Näppäile seuraava rivi, ennen kuin alat tutkia ikkunoita:

```
KEY 139,chr$(13)+"mode 2:paper 0:ink 1,0
:ink 0,9:list"+chr$(13)
```

Näillä asetuksilla pienempi ENTER-näppäin tyhjentää kuvaruudun ja palauttaa tekstin näkyviin käyttäen INK-käskyissä määriteltyjä värejä. Voit käyttää tätä apunasi, jos sekoat PEN- ja PAPER-käskyjen väriyhdistelmissä kadottaen tekstit näkyvistä. Seuraava ohjelma piirtää kuvaruutuun joukon ikkunoita ja esittelee kahta tärkeää seikkaa:

```
5 MODE 0
10 FOR n=0 TO 7
20 WINDOW #n,n+1,n+6,n+1,n+6
30 PAPER #n,n+4
40 CLS #n
50 FOR c=1 TO 200:NEXT
60 NEXT
```

Ensimmäinen huomion arvoinen seikka on se, että kukin uusi ikkuna kirjoittuu edellisen päälle. Toinen tärkeä seikka on se, että viestit ilmestyvät tulostusreitille nro 0 (#0) joka kerta, ellei sitä ohjata erikseen jollekin toiselle tulostusreitille. Ajettuasi ohjelman RUN-käskyllä näppäile LIST-käsky ennen kuin teet mitään muuta. Ja kuten huomaat, on ohjelmalistaus ahdettu ikkunaan 0. Näppäile seuraavaksi LIST #5 ja sen jälkeen CLS #6.

Tästä näkyy, että viimeksi määritelty ikkuna sijoittuu aikaisempien päälle, ja että **Ready** -ilmoitus tulostuu ikkunaan nro 0, vaikka listaus pyydettiinkin ikkunaan nro 5. Lisää ohjelmaan rivi 55, jossa käytetään WINDOW SWAP-käskyä:

```
55 IF n=3 THEN WINDOW SWAP 7,0
```

Voisi kuvitella, että nyt ohjelman päätyttyä saatava **Ready** -ilmoitus ohjautuisi tulostusreitille 7 (ikkunaan 7). Aja ohjelma RUN-käskyllä niin näet miten siinä käy. Kokeile CLS- ja LIST-käskyjä eri WINDOW SWAP-arvojen (7,0) kanssa. Myös WINDOW- ja WINDOW SWAP-käskyjä käsitellään lisää luvussa 8.

Voit kehitellä edellä olevaa yksinkertaista esimerkkiohjelmaa edelleen niin näet, millä tavalla ikkunat toimivat.

LUKU 6: ÄÄNEN ALKEET

Tässä luvussa käsiteltäviä aiheita ovat:

- Säveljaksot
- SOUND-käsky
- Vaipat
- Äänikäskyjen laittaminen jonoon ja synkronointi eli tahdistus

CPC 464:ssä on oma kaiutin, jonka kautta tietokoneen muodostamat äänitehosteet kuuluvat. Jos laitteistoosi kuuluu modulaattori/virtalähdeyksikkö MP1 ja tavallinen TV, säädä television äänenvoimakkuus aivan hiljaiselle.

Äänen voimakkuutta voidaan säätää tietokoneen oikeassa päädyssä olevalla äänenvoimakkuuden säätimellä (VOLUME). Voit myös syöttää CPC 464:n muodostamat äänet stereolaitteistosi tuloliitäntään, jossa on merkintä "AUX". Liitäntä tapahtuu CPC 464:n takapaneelin vasemmassa reunassa olevasta tulo/lähtöliitännästä (I/O). Näin voit kuunnella CPC 464:n muodostamia stereoääniä stereofonisesti HiFi-kaiuttimien tai kuulokkeiden kautta.

Yksinkertaisimmillaan äänen muodostaminen tapahtuu käskyllä PRINT CHR$(7).

Tämä saa aikaan yksittäisen piipahduksen, mutta haluat varmasti käyttöösi CPC 464:n monipuoliset äänitehosteet. Niinpä käsittelemmekin äänten muodostamista ensin yleisellä tasolla, ja perehdymme sitten äänikäskyissä käytettäviin avainsanoihin hieman tarkemmin. Niillä eväillä voit kokeilla sävelasteikon muodostamista ja tutustua erilaisten soitinten ääniin.

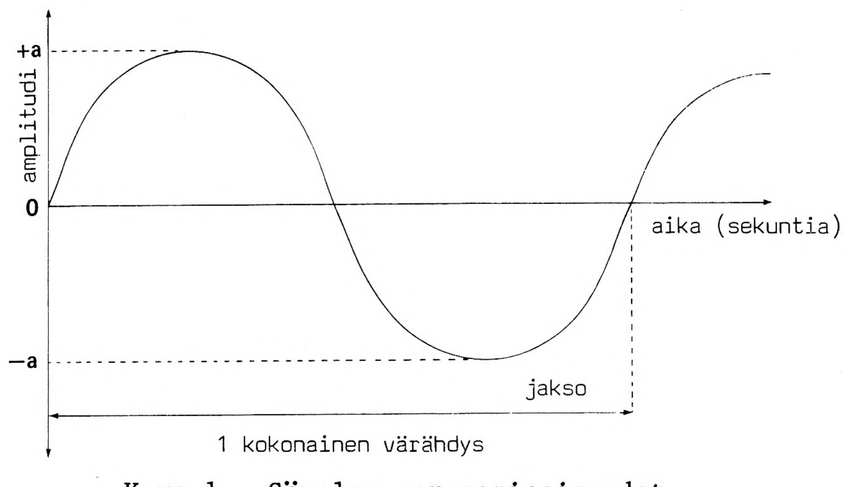
6.1 ÄÄNEN PERUSKÄSITTEITÄ

Tarkastelkaamme aluksi hieman lähemmin esim. jonkin soittimen muodostamaa ääntä. Yhteen yksittäiseen säveleen liittyy seuraavia ominaisuuksia:

1) Sävelkorkeus ja sävelkorkeuden vaihtelut sävelen keston aikana.
2) Sävelen voimakkuus ja voimakkuusvaihtelut sävelen keston aikana.
3) Sävelen kesto eli pituus.

6.2 SÄVELKORKEUS

Sävelkorkeus on varmasti kunkin sävelen tärkein ominaisuus. Korvaan kantautuva sävel on säännöllistä värähtelyä, ja kaikilla värähtelyillä on taajuus, jakso ja amplitudi (katso kuvaa 1).



Kuva 1. Sävelen perusominaisuudet

Taajuus on luku, joka ilmoittaa värähdysten määrän sekunnissa. Jakso on taas kunkin värähdyksen pituus sekunneissa. Amplitudi kuvaa puolestaan äänen voimakkuutta, mutta jätämme sen tässä käsittelemättä ja keskitymme sävelkorkeuteen.

Taajuus ja jakso liittyvät toisiinsa seuraavan yksinkertaisen kaavan kautta:

Taajuus [hertseissä (Hz)] = 1/jakso [sekunneissa (s)]

SOUND-käskyssä olevan "säveljakson" ja taajuuden välinen suhde on seuraava:

Taajuus hertseissä (Hz) = 125000/säveljakso

Jos siis ´säveljakso´ on 1000, muodostaa CPC 464 äänen, jonka taajuus on 125 Hz. Vastaavasti ´säveljakso´ 25 tuottaa 1000 Hz:n (1 kHz:n) äänen.

Sävelkorkeuden määrittämiseen voitaisiin käyttää kumpaa hyvänsä edellä mainituista käsitteistä, mutta AMSTRADin BASIC-kieleen on valittu käsite ´säveljakso´. Paina mieleesi, että sävelkorkeus laskee säveljakson lukuarvon kasvaessa. SOUND-käskyn osa ´säveljakso´ on siis se, jolla sävelkorkeutta muutetaan. SOUND-käskyssä käytettävän säveljakson lukualue on 0 - 4095, ja sen täytyy olla aina kokonaisluku. Tämä saattaa aiheuttaa sen, että eräät sävelet eivät osu täsmälleen oikeisiin kohtiin sävelasteikolla. Ero on kuitenkin niin pieni, että siitä tuskin on mitään haittaa. Sävelkorkeudet ja vastaavat säveljaksojen lukuarvot on lueteltu liitteessä VII.

Sävelkorkeus voi vaihdella, kun säveliä muodostetaan oikeilla soittimilla. Sävelkorkeuden vaihtelu saadaan aikaan yleensä vibraatto-tehosteella. Kunkin sävelen jakson rakennetta on mahdollista muutella ENT-käskyllä. ENT-käsky vaikuttaa ns. sävelvaippaan, joka voidaan sisällyttää SOUND-käskyyn.

6.3 VOIMAKKUUS

Voimakkuus ilmoittaa yksinkertaisesti sen, kuinka lujaa ääni kuuluu. CPC 464:n muodostaman äänen perusvoimakkuus voidaan asettaa SOUND-käskyn osalla ´voimakkuus´. Sen lukuarvo voi olla väliltä 0 - 15, ja senkin on oltava aina kokonaisluku. Äänenvoimakkuus on sitä suurempi, mitä suurempi on sen määrittämiseen käytetty luku.

Jos olet jo kokeillut yksinkertaisten sävelten muodostamista CPC 464:llä olet jo varmaan havainnut, miten yksitoikkoisia perusäänet ovat, ellei niitä mitenkään "käsitellä". Eräs syy on se, että tavallisilla soittimilla soitettaessa äänenvoimakkuus kasvaa sävelen alussa. Vastaavasti äänenvoimakkuus laskee sävelen lopussa. Näitä alussa tapahtuvaa "syttymistä" ja lopussa tapahtuvaa "sammumista" voidaan jäljitellä monin eri tavoin tietokoneen muodostaessa ääniä. CPC 464:ssä tähän käytetään ENV-käskyä ja siihen tarkemmin liittyvää käsitettä ´voimakkuusvaippa´. Voimakkuusvaippa vaikuttaa juuri äänenvoimakkuuden muotoon esim. sävelen alussa ja lopussa.

6.4 SÄVELEN KESTO ELI PITUUS

Sävelen kesto on myös eräs sävelen perusominaisuuksia, ja sävelen keston perusyksikkö on neljännesnuotti. Lisäksi puhutaan puolinuoteista, kuudestoistaosanuoteista, jne., mutta ne saadaan perusyksiköstä kertomalla tai jakamalla. Säveliä voidaan kuitenkin soittaa erilaisilla nopeuksilla, joten lienee syytä määritellä sävelen pituuden perusyksikkö, jotta eripituisia ääniä voitaisiin muodostaa.

CPC 464:ssä tähän käytetään SOUND-käskyn osaa 'kesto', joka on kokonaisluku. Yksi yksikkö = 1/100 sekuntia (eli 100 yksikköä = 1 sekunti). Sävelen kesto voidaan myös määritellä 'sävelvaipan' pituudella, mikä helpottaa sävelten muodostamista 'sävelvaippaa' käytettäessä.

6.5 MUITA ÄÄNIÄ

Tietokoneella voidaan muodostaa myös ns. valkoista kohinaa, joka sekin on ääntä. Sitä ei kuitenkaan voida käyttää yksittäisten sävelten muodostamiseen, mutta sitä voidaan käyttää sävelten ja äänten taustana. Näin saadaan muodostetuksi erilaisia tehosteita. Myös pelkkää kohinaa voidaan käyttää esim. räjähdysäänien muodostamiseen, joka on yleensä saatu aikaan valkoisella kohinalla, jonka voimakkuusvaippaa on säädetty. Kohinan taajuus vaihtelee sattumanvaraisesti. Kohina määritellään SOUND-käskyn 'kohinajakso' osalla. Eräs CPC 464:n käyttöön liittyvä tarkeä seikka on, että vaikka käytettävissä onkin kolme erillistä äänikanavaa, joihin kuhunkin voidaan erikseen määritellä omat 'säveljaksonsa', voidaan kohinan muodostamiseen käyttää vain yhtä 'kohinajaksoa' kerrallaan.

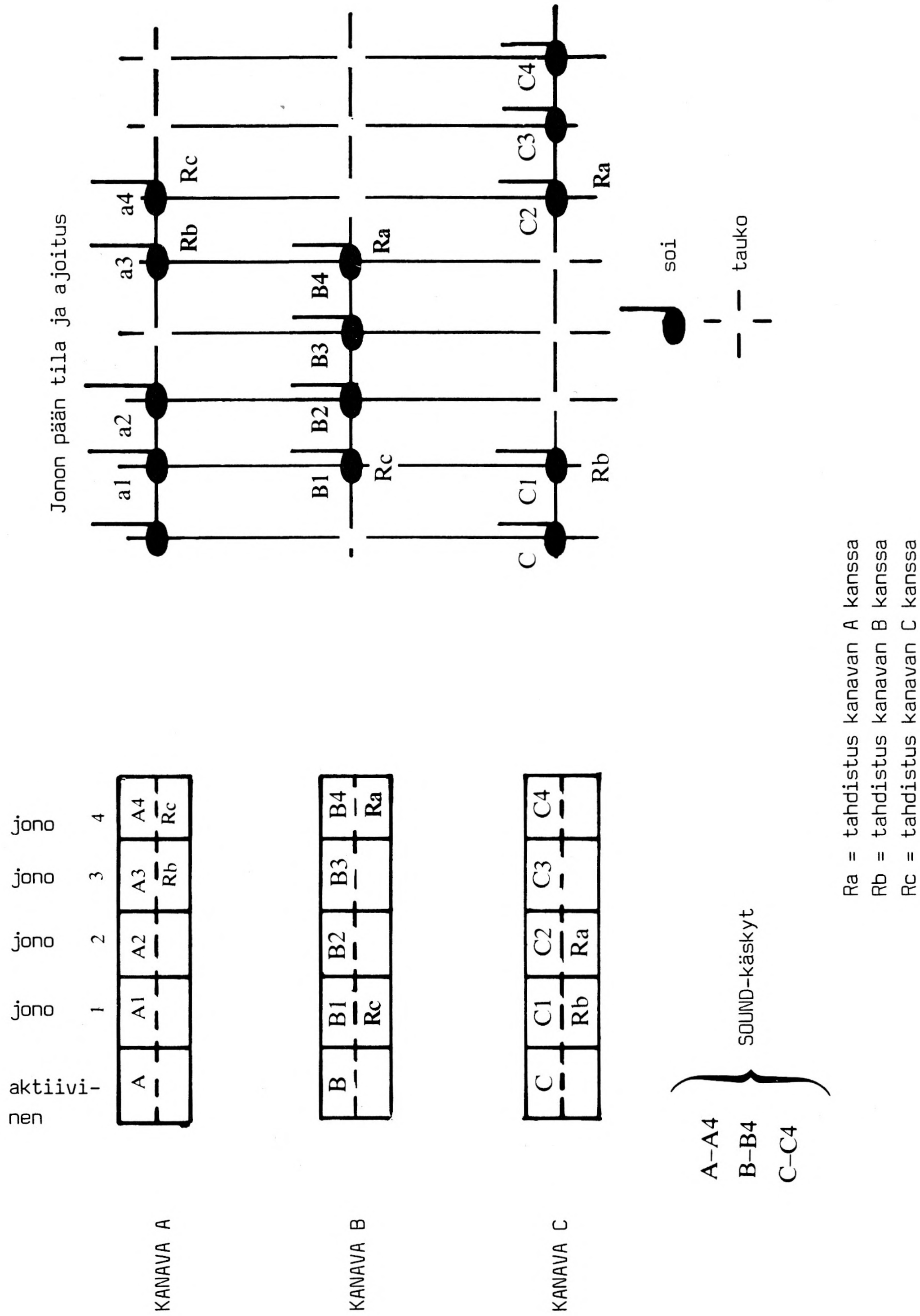
6.6 MONIÄÄNISYYS JA ÄÄNIKANAVAT

Useimpien musiikkikappaleiden nuoteissa esiintyy ainakin kahdenlaisia nuottiavaimia, nimittäin basso- ja diskanttiavaimia. Jotta myös CPC 464:llä voitaisiin tuottaa musiikkia ainakin näille kahdelle nuottiavaimelle, on CPC 464:ssä kolme äänikanavaa: kanavat A, B ja C. Kukin kanava voi soittaa musiikkia toisistaan riippumatta, tai ne voidaan tahdistaa keskenään tarpeen mukaan. Kanavat valitaan SOUND-käskyn osalla 'kanavan tila'.

6.7 KANAVIEN ÄÄNIJONOT

Jokaiseen kolmeen kanavaan voidaan järjestää soitettavia ääniä jonoon. Jonossa on tilaa viidelle erilliselle SOUND-käskylle, joista yksi on aktiivinen eli parhaillaan käytössä ja neljä odottamassa vuoroaan. CPC 464:n käyttöjärjestelmä voi jatkaa muiden tehtävien suorittamista soittaessaan jonossa olevia ääniä, ja se voi palata tarvittaessa takaisin laittamaan lisää SOUND-käskyjä jonoon sitä mukaa kun jonoon tulee tilaa.

ÄÄNIKANAVIEN TAHDISTUSKOHDAT



6.8 KANAVAN TILA

Haluttaessa tietää jonkin kanavan tila, käytetään avainsanaa SQ. Samalla avainsanalla saadaan tietoja jonossa olevista vapaista paikoista sekä tahdistuskohdista ("RENDEZVOUS") ja pidoista (HOLDS). ON SQ GOSUB on keskeytyskäsky, jota käytetään kiinnittämään tietokoneen huomio takaisin ohjelman siihen osaan, jossa äänet muodostetaan.

6.9 TAHDISTUSKOHTA (RENDEZVOUS) JA PITO (HOLD)

CPC 464:ssä on tahdistusmahdollisuus, jolla eri kanavat voidaan tahdistaa eli synkronoida keskenään. Tahdistuskohtaan laitetaan merkki kahdelle tai useammalle kanavalle, jolloin kyseisten kanavien SOUND-käskyt suoritetaan samanaikaisesti. Ne ovat varsin käyttökelpoisia tahdistettaessa toisiinsa jatkuvia säveliä, jotka sisältävät sellaisia katkoja tai taukoja, jotka eivät esiinny samanaikaisesti molemmilla kanavilla.

Pitotoiminto (HOLD) (katso kohtaa 6.10.1) on tärkeä silloin, kun halutaan tahdistaa kanavat keskenään kokonaisvaltaisesti (eli heti sävelen alussa). Samassa yhteydessä voidaan käyttää viivettä, ja äänijonojen on oltava valmiina jo tässä vaiheessa. Tällaisessa tilanteessa käytetään HOLD- ja RELEASE-avainsanayhdistelmää.

6.10 SOUND-KÄSKYN OSAT

SOUND-käsky koostuu seuraavista osista: SOUND G,H,I,J,K,L,M

Jossa:
G = kanavan tila
H = säveljakso
I = kesto
J = voimakkuus
K = voimakkuusvaippa
L = sävelvaippa
M = kohinajakso

SOUND-käskyn parametrit G - M ovat kaikki kokonaislukuja, ja vain kaksi ensimmäistä ovat pakollisia. Loput viisi ovat valinnaisia eli niitä voidaan haluttaessa käyttää tai ne voidaan jättää pois, mutta niillä kaikilla on oletusarvot eli arvot, jotka tietokone valitsee ellei mitään muuta ole määritelty. Näitä asioita käsitellään tarkemmin seuraavissa kappaleissa.

6.10.1 SOUND-käskyn parametrien kuvaus

<u>G: Kanavan tila</u>

G:n arvo voi olla väliltä 1...255.
Tietokoneen valitsema oletusarvo, jos käyttäjä ei määrittele sitä: ei oletusarvoa (pakko määritellä).

CPC 464:llä on mahdollista soittaa jopa kolme eri SOUND-käskyllä määriteltyä ääntä samanaikaisesti. Tämä on saatu aikaan käyttämällä kolmea kanavaa, joita nimitetään kanaviksi A, B ja C. G:n arvo annetaan kokonaislukuna desimaalimuodossa, mutta kun tietokone muuttaa sen 8-bittiseksi arvoksi (jolloin saadaan esille vähiten merkitsevä ja eniten merkitsevä bitti), määräytyy bittien perusteella seuraavat komennot:

Desimaalimuoto	Bitti	Komento
1	0 LSB	Lähetä ääni kanavaan A
2	1	Lähetä ääni kanavaan B
4	2	Lähetä ääni kanavaan C
8	3	Tahdistus kanavan A kanssa
16	4	Tahdistus kanavan B kanssa
32	5	Tahdistus kanavan C kanssa
64	6	Pito (HOLD)
128	7 MSB	Tyhjennys (FLUSH)

(LSB = Least Significant Bit = vähiten merkitsevä bitti, ja MSB = Most Significant Bit = eniten merkitsevä bitti).

Kanavan tilan esimerkkiarvoja:

2 = lähetä seuraava ääni lähtökanavaan B.
5 = lähetä seuraava ääni lähtökanaviin A ja C.
98 = 64 + 32 + 2
= lähetä seuraava ääni lähtökanavaan B, tahdistus kanavan C kanssa ja pito.

On tärkeää muistaa, että jos tahdistus halutaan järjesteää kahden (tai useamman) kanavan kesken, on tarpeen liputtaa kunkin kanavan ´kanavan tila´ sopivalla hetkellä.

Jos pitoa (HOLD) käytetään minkä hyvänsä äänikanavien yhdistelmän kanssa, se pysäyttää kyseisen käskyn käsittelyn ja pysäyttää sen takana olevan jonon, kunnes se vapautetaan RELEASE-käskyllä (tai "tyhjennetään" jäljempänä tulevassa SOUND-käskyssä olevalla tyhjennystoiminnolla (FLUSH)).

Kun tyhjennysbitti (FLUSH) aktivoidaan, suoritetaan tyhennysbitin sisältävä SOUND-käsky välittömästi. Samalla jono tyhjennetään ja jonon pää asetetaan ei-aktiiviseksi. Samalla loppuu minkä hyvänsä parhaillaan soivan äänen muodostus.

H: Säveljakso

H:n arvo voi olla väliltä 0...4095.
Tietokoneen valitsema oletusarvo, jos käyttäjä ei määrittele sitä: ei oletusarvoa (pakko määritellä).

Tällä numeroarvolla valitaan säveljakso, joka puolestaan määrää soitettavan äänen taajuuden (sävelkorkeuden). Taajuus voidaan laskea kaavasta:

Taajuus = 125000/jakso

Jos halutaan käyttää pelkkää kohinaa, on kätevintä antaa H:lle arvo 0.

I: Kesto

I:n arvo voi olla mikä hyvänsä kokonaisluku väliltä -32768...+32767.
Tietokoneen valitsema oletusarvo, jos käyttäjä ei määrittele sitä: 20.

Nollaa suuremmat I:n arvot vastaavat 1/100 sekunteja. Jos I:n arvoksi annetaan 0, määräytyy kesto sen mukaan, kuinka pitkäksi voimakkuusvaippa on määritelty.

Jos I:n arvoksi määritellään jokin nollaa pienempi eli negatiivinen luku, määrää luvun positiivinen arvo niiden kertojen lukumäärän, jonka erikseen määritelty voimakkuusvaippa on toistettava.

J: Voimakkuus

J:n arvo voi olla mikä hyvänsä kokonaisluku väliltä 0...15 (J voi olla 0...7, jos voimakkuusvaippaa ei ole määritelty).
Tietokoneen valitsema oletusarvo, jos käyttäjä ei määrittele sitä: 12 (4, jos voimakkuusvaippaa ei ole määritelty).

J määrittelee sävelen alkuvoimakkuuden, ja voimakkuutta voidaan muuttaa voimakkuusvaipalla, mikäli se on määritelty. J:n arvo 0 tarkoittaa, ettei ääni kuulu lainkaan.

K: Voimakkuusvaippa

K:n arvo voi olla mikä hyvänsä kokonaisluku väliltä 0...15.
Tietokoneen valitsema oletusarvo, jos käyttäjä ei määrittele sitä: 0.

K:n arvo määrittelee erikseen määriteltävän voimakkuusvaipan numeron. Vaipan määrittelyyn käytetään ENV-käskyä. Voimakkuusvaipan lukuarvo 0 on kiinteä arvo, eikä sitä voida muuttaa ENV-käskyllä. Tällöin arvo on kiinteä 2 sekuntia, ja voimakkuus määräytyy J:n arvon mukaan.

L: Sävelvaippa

L:n arvo voi olla mikä hyvänsä kokonaisluku väliltä 0...15. Tietokoneen valitsema oletusarvo, jos käyttäjä ei määrittele sitä: 0.

L:n arvo määrittelee erikseen määriteltävän sävelvaipan numeron. Sävelvaipan määrittelyyn käytetään ENT-käskyä. Sävelvaipan lukuarvo 0 on kiinteä arvo, jota ei voida muuttaa ENT-käskyllä. Arvo määräytyy säveljakson H mukaan.

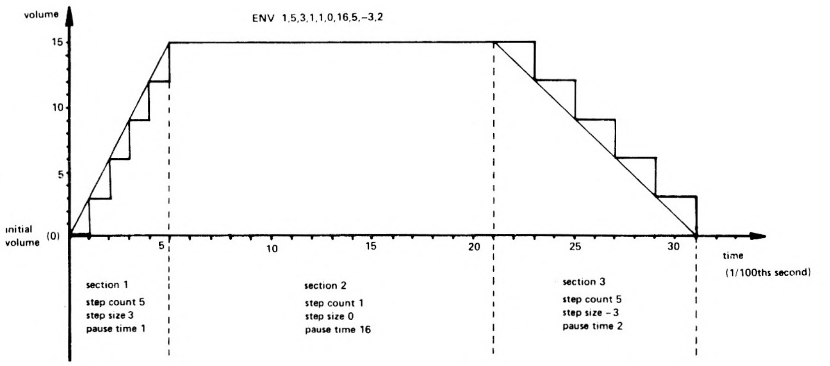
M: Kohinajakso

M:n arvo voi olla mikä hyvänsä kokonaisluku väliltä 0...15. Tietokoneen valitsema oletusarvo, jos käyttäjä ei määrittele sitä: 0.

M:n arvo määrittelee SOUND-käskyllä määriteltävään ääneen lisättävän kohinan. Kohinaa ei lisätä ääneen, jos arvoksi annetaan 0 (joka on myös samalla oletusarvo). Muista, että kerrallaan voidaan asettaa soimaan vain yksi kohinajakso. Tämä tarkoittaa sitä, että sama kohina kuuluu kaikilta niiltä kanavilta, joille on määritelty kohinaa.

6.11 ENV-KÄSKY JA VOIMAKKUUSVAIPAT

Sävelet saadaan elävämmiksi käyttämällä niissä alussa voimistuvaa ja lopussa vaimentuvaa äänenvoimakkuutta. Tämä saadaan aikaan ENV-käskyllä, jota käytetään sävelen voimakkuusvaipan muokkaamiseen. ENV-käskyn rakenne kannattaa ensin hahmotella paperille, ennen kuin sitä yritetään syöttää tietokoneeseen numeromuodossa. Katsotaanpa erästä esimerkkiä:



section	=	osa
step count	=	portaiden määrä
step size	=	portaan koko
pause time	=	portaan kesto
initial volume	=	alkuvoimakkuus
volume	=	voimakkuus
time	=	aika

Kuva 3. Esimerkki voimakkuusvaipasta

Voimakkuusvaipan muoto täytyy muuttaa numeroarvoiksi, jotta se voidaan syöttää tietokoneelle käskylausekkeessa. Paperille hahmoteltu voimakkuusvaipan muoto jaetaan ensin osiin, joita voi olla enintään viisi. Kunkin osan on kuitenkin koostuttava suorasta viivasta. Seuraavaksi kukin osa jaetaan portaisiin, ja valitsemiesi portaiden lukumäärää nimitetään ´portaiden määräksi´. Portaiden pituus voidaan puolestaan määritellä ´portaan kestolla´, jossa 1 yksikkö vastaa 1/100 sekuntia.

Myös portaan amplitudi eli koko voidaan määritellä. Ääni voimistuu, jos portaan koko määritellään positiivisella luvulla. Vastaavasti ääni heikkenee, jos portaan koko määritellään negatiivisella luvulla. Jos johonkin osaan ei määritellä lainkaan portaita, pysyy voimakkuus vakiona seuraavaan osaan asti.

HUOM!
Yritettäessä jäljitellä soittimien ääniä tilanne on usein sellainen, että kunkin sävelen voimakkuus on nolla sekä alussa että lopussa. Siksipä SOOUND-käskyllä määriteltävän äänen alkuvoimakkuus pitäisikin aina määritellä nollaksi, jonka jälkeen äänenvoimakkuutta säädetään voimakkuusvaipalla.

Käskyn muoto:

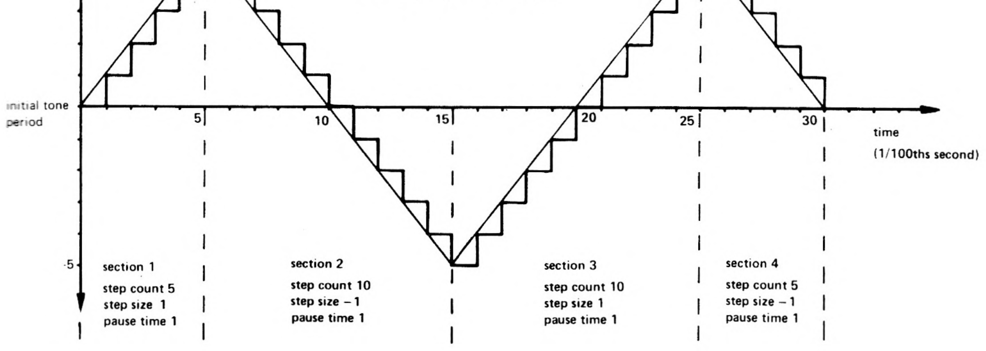
ENV N,P1,Q1,R1,P2,Q2,R2,P3,Q3,R3,P4,Q4,R4,P5,Q5,R5

N	= vaipan numero	väliltä 1...15
P1 - 5	= portaiden määrä (osat 1 - 5)	väliltä 0...127
Q1 - 5	= portaan koko (osat 1 - 5)	väliltä -128...+127
R1 - 5	= portaan kesto (osat 1 - 5)	väliltä 0...255

ENV-käskyyn on aina pakko määritellä ´vaipan numero´. Samoin käskystä on aina löydyttävä yksi kokonainen osa (section), mutta osien lukumäärä on valinnainen. Jos esimerkiksi osat 4 ja 5 jätetään pois, on käskyssä kolme osaa.

6.12 ENT-KÄSKY JA SÄVELVAIPAT

Rakenne on täsmälleen samanlainen kuin voimakkuusvaipoillakin, mutta sävelvaipalla pystytään muuttamaan sävelen taajuutta eli sävelkorkeutta. Näin saadaan siis aikaan eräänlainen vibraatto. Kun olet päättänyt mielessäsi, minkä muotoinen sävelen taajuuden pitää olla, hahmottele se paperille samalla tavalla kuin voimakkuusvaippakin ja jaa se osiin ja portaisiin esimerkiksi alla olevan esimerkin tapaan:



section	= osa
step count	= portaiden määrä
step size	= portaan koko
pause time	= portaan kesto
tone period	= säveljakso
initial tone period	= alku-säveljakso
time	= aika

Kuva 4. Sävelvaipan eri osat

Suurin ero voimakkuusvaipan ja sävelvaipan välillä on se, ettei sävelvaipalla ole mitään vaikutusta sen enempää SOUND-käskyssä kuin voimakkuusvaipassakaan aiemmin määritellyn sävelen kestoon. Jos sävelvaippa loppuu ennen säveltä, niin SOUND-käskystä saatava säveljakson vakioarvo on se, joka määrää jäljellä olevan sävelen osan taajuuden. Jos sävelvaippa on taas pitempi kuin sävelen kesto, niin ylittävä osa jätetään huomioimatta.

Jos sävelvaippaa halutaan käyttää uudelleen kyseisen sävelen kestäessä, käytetään negatiivista vaipan numeroa (huom. sitä ei kutsuta negatiivisena SOUND-käskyssä).

Käskyn muoto:

ENT S,T1,V1,W1,T2,V2,W2,T3,V3,W3,T4,V4,W4,T5,V5,W5

S = vaipan numero	väliltä 1...15 (vaippa saadaan toistumaan negatiivisella arvolla)
T1 - 5 = portaiden määrä (osat 1 - 5)	väliltä 0...239
V1 - 5 = portaan koko (osat 1 - 5)	väliltä -128...+127
W1 - 5 = portaan kesto (osat 1 - 5)	väliltä 0...255 (1/100 sekuntia)

S on pakollinen, ja kunkin yksittäisen osan (section) on oltava täydellinen.

Kun sävelvaipan numero määritellään, mitätöityvät kaikki aikaisemmat asetukset. Määriteltäessä kumpaa hyvänsä tyyppiä oleva vaippa jossa ei ole yhtään osaa, nollautuvat kyseisen vaipan asetukset nollaksi.

6.13 MUITA AIHEESEEN LIITTYVIÄ TOIMINTOJA JA KÄSKYJÄ

<u>Käsky SQ(x)</u>

x tarkoittaa kanavan numeroa, ja se voi olla 1, 2 tai 4 vastaten kanavia A, B ja C. (Katso sivulla 124 olevaa taulukkoa, jossa käsiteltiin SOUND-käskyn kanavan tilan bittiarvoja). Käskyllä SQ(x) voidaan kysyä nimetyn äänikanavan tilaa. Vastaukseksi saadaan jokin kokonaisluku, joka on väliltä 0...255. Tieto saadaan ulos lukua vastaavan binäärimuodon biteistä seuraavasti:

Bitti	
0 - 2	testattavassa jonossa olevien vapaiden paikkojen lukumäärä (arvo 0 - 4).
3	tahdistus A-kanavan kanssa jonon päässä
4	tahdistus B-kanavan kanssa jonon päässä
5	tahdistus C-kanavan kanssa jonon päässä
6	pito (HOLD) jonon päässä
7	parhaillaan käytössä oleva kanava

Bittejä 3 - 6 (tahdistusta ja pitoa) kuvattiin jo kanavan tilaa käsittelevässä luvussa. Lisäksi tällä testillä voidaan estää ON SQ GOSUB-käsky, jota käsitellään seuraavassa kappaleessa.

Käsky ON SQ GOSUB

Käsky on muotoa ON SQ(y) GOSUB´rivinumero´

y tarkoittaa kanavan numeroa, ja se voi olla 1, 2 tai 4 vastaten kanavia A, B ja C.

Tämä on "viritetty" keskeytyskomento, ja sen avulla voidaan tutkia, onko nimetyssä äänijonossa rakoa. Raon havaitessaan käsky "laukaisee" toiminnan kohdassa ´rivinumero´ määriteltyyn aliohjelmaan. Sekä SQ- että SOUND-avainsana poistavat keskeytyksen. Aliohjelma päätetään RETURN-käskyllä tavalliseen tapaan. Kaikilla kanavilla on sama prioriteetti (etuoikeus), ja kun nimetyssä kanavassa on tilaa, käsky "laukeaa" purkaen samalla itsensä. Täten aliohjelman on palautettava keskeytys, mikäli sitä tarvitaan uudelleen.

Käsky RELEASE

Käsky on muotoa RELEASE z. z tarkoittaa tässä kanavanumeroa (tai kanavanumeroita) ja sen arvo voi olla jokin kokonaisluku väliltä 1 - 7.

Tietyn kanavan tiettyyn SOUND-käskyyn voidaan liputtaa HOLD-toiminto (pito), kuten jo aikaisemmin mainittiin SOUND-käskyn ´kanavan tilaa´ käsittelevässä osassa. RELEASE-käskyä käytetään yksinkertaisesti näiden pitotoimintojen poistamiseen. Vapautuksen kohteena oleva kanava (tai kanavat) määräytyvät annetun luvun binäärimuodon bittien perusteella:

Bitti 0 = kanava A
Bitti 1 = kanava B
Bitti 2 = kanava C

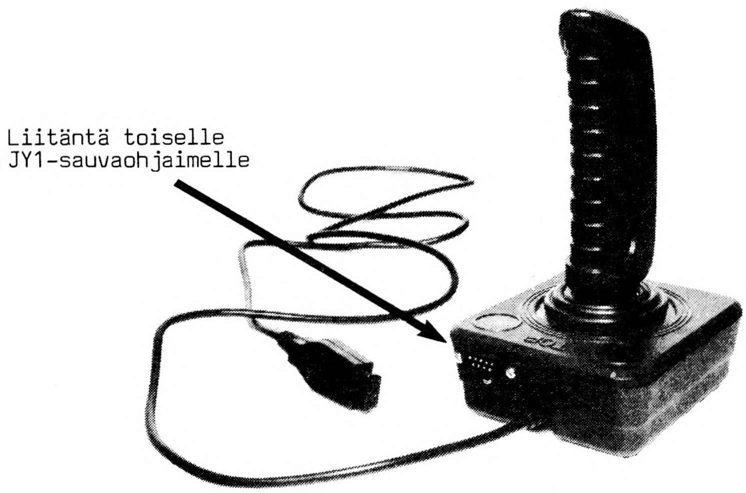
Jos RELEASE-käsky kohdistetaan sellaiseen kanavaan jossa ei ole pito-toimintoa (HOLD), sillä ei ole mitään vaikutusta ohjelman suoritukseen.

LUKU 7: KIRJOITTIMET JA SAUVAOHJAIMET

Tässä luvussa käsitellään seuraavia asioita:

- Sauvaohjaimet (Joystick)
- Kirjoittimet
- Laitteiden liitäntä CPC 464:ään

CPC 464:ssä on valmis Centronics-tyyppinen liitäntä kirjoitinta varten. Samoin CPC 464:ään voidaan liittää suoraan yksi tai kaksi sauvaohjainta. Voit hankkia lisävarusteena esim. sauvaohjaimen JY1, jos pelaat CPC 464:llä sellaisia pelejä, joissa on mahdollisuus käyttää sauvaohjainta ja joissa esiintyy "tulitusta" muodossa tai toisessa. Sauvaohjain JY1 voidaan liittää suoraan CPC 464:n takana olevaan 9-napaiseen liitäntään, jossa on merkintä "USER PORTS (I/O)". CPC 464:n kanssa voidaan käyttää kahta sauvaohjainta. Toinen sauvaohjain liitetään ensimmäisessä sauvaohjaimessa olevaan liitäntään. Liittimien kytkennät löytyvät käsikirjan lopusta liitteestä V, joissa kuvataan myös muiden liitäntöjen kytkentää.



Sauvaohjain JY1

7.1 SAUVAOHJAIMET

AMSTRAD CPC 464:n ohjelmisto on tehty sellaiseksi, että se toimii joko yhdellä tai kahdella sauvaohjaimella. Ohjelma käsittelee sauvaohjaimia ikäänkuin ne olisivat osa näppäimistöä, ja sauvaohjainten tilaa voidaan tunnustella ohjelmista INKEY$- ja INKEY-käskyillä. Jos sauvaohjaimessasi on vain yksi tulituspainike, se on mitä todennäköisemmin "tulituspainike 2" CPC 464:n terminologiassa.

CPC 464:ssä on erikoistoiminto, jonka avulla sauvaohjainten tilaa voidaan tutkia myös suoraan. Siihen käytetään toimintoa JOY(0) sauvaohjaimelle I ja JOY(1) sauvaohjaimelle II. Toiminnolla saadaan selville sauvaohjaimen kytkinten tila samalta kierrokselta, jolloin näppäimistön näppäinten tilaa on viimeksi tutkittu. Näppäimistön näppäinten tilaa tutkitaan 50 kertaa sekunnissa, joten sauvaohjainten kytkinten tila tiedetään käytännöllisesti katsoen joka hetki.

Sauvaohjainten tilaa tutkitaan JOY(0):n ja JOY(1):n bittien avulla alla olevan taulukon osoittamalla tavalla. Sarakkeen "näppäimen numero" arvoa käytetään INKEY-käskyn yhteydessä, ja sarakkeessa "Vastaava näppäin" on sauvaohjaimen II toimintoja vastaavat näppäimistön näppäimet.

Sauva-ohjain I	JOY(0)	Näppäimen numero	Sauva-ohjain II	JOY(1)	Näppäimen numero	Vastaava näppäin
Ylös	Bitti 0	72	Ylös	Bitti 0	48	6
Alas	Bitti 1	73	Alas	Bitti 1	49	5
Vasemmalle	Bitti 2	74	Vasemmalle	Bitti 2	50	R
Oikealle	Bitti 3	75	Oikealle	Bitti 3	51	T
Tulitus 2	Bitti 4	76	Tulitus 2	Bitti 4	52	G
Tulitus 1	Bitti 5	77	Tulitus 1	Bitti 5	53	F

Sauvaohjaimen II tilaa tutkittaessa CPC 464 ei pysty erottamaan, halutaanko tietää sauvaohjaimen II kytkinten tilat vaiko vastaavien näppäimistön näppäinten tilat. Käytännössä tämä tuskin aiheuttaa hämmennystä, mutta näppäimistön vastaavia näppäimiä tosiaan voidaan käyttää sauvaohjaimen II sijasta.

AMSOFTin sauvaohjainta JY1 käytettäessä myös sauvaohjain II on samanlainen kuin sauvaohjain I. Sauvaohjain II liitetään sauvaohjaimen I runko-osassa olevaan liitäntään. Muita liitäntöjä tai kytkentöjä ei tarvitse tehdä sauvaohjaimen II käyttämiseksi.

CPC 464:n 9-napaiseen käyttäjänporttiin (USER PORTS (I/O)) voidaan liittää myös muihin mikrotietokoneisiin sopivia vakiomallisia sauvaohjaimia. Niihin ei kuitenkaan voi liittää sauvaohjainta II ilman erityistä sovitinta. Sinun ei kuitenkaan kannata yrittää käyttää tällaista sauvaohjainta AMSOFTin sauvaohjaimen JY1 rungossa olevaan liitäntään kytkettynä sauvaohjaimena II.

Jos teet itse omia ohjelmia, voit laittaa ohjelman alkuun mahdollisuuden valita joko sauvaohjainten käytön tai kohdistimen siirtonäppäinten käytön, jolloin esim. [COPY]-näppäintä tai muuta näppäintä voidaan käyttää tulitukseen.

7.2 KIRJOITTIMEN LIITÄNTÄ

CPC 464:ssä on valmis Centronics-tyyppinen liitäntä kirjoittimelle.

Kirjoittimen liitosjohto on rakenteeltaan "yksi yhteen" CPC 464:n liittimen ja kirjoittimessa olevan liittimen välillä. CPC 464:n liitinkortissa on kaksi liitäntäliuskaa vähemmän kuin kirjoittimen liittimessä, jolloin liitäntään voidaan käyttää vakiomallista korttiliitintä.

Kirjoitinliitännän yksityiskohtaiset tiedot löytyvät liitteestä V.

Liitosjohto kytketään siten, että CPC 464:n nasta 1 kytketään kirjoittimen nastaan 1, CPC 464:n nasta 19 kytketään kirjoittimen nastaan 19, jne. Kirjoittimen nastoja 18 ja 36 ei kytketä CPC 464:ään.

Huomaa, että CPC 464:n liitinkortin alemmat liitinliuskat on numeroitu 19:stä eteenpäin (eikä 18:sta eteenpäin, kuten saattaisi luulla, koska ylemmässä rivissä on 17 liitäntäliuskaa). Tämä siksi, että jokainen käytettävä johdin kytketään täsmälleen samannumeroiseen CPC 464:n liitinnastaan ja kirjoittimen liitinnastaan.

CPC 464 käyttää BUSY-signaalia (nasta 11) tietokoneen ja kirjoittimen väliseen synkronointiin (tahdistukseen), ja CPC 464 odottaa, jos kirjoitin on "OFF LINE"-tilassa.

Käyttäjän ei tarvitse tehdä mitään asetuskäskyjä, ja tulostus ohjataan kirjoittimelle määrittelemällä tietokoneelta tulostusreitti 8 seuraavasti:

```
LIST #8
```

Tällä käskyllä ohjelmamuistissa olevan BASIC-ohjelman listaus tulostuu kirjoittimella - kunhan ohjelma on vain tyypiltään sopiva listattavaksi eli suojaamaton.

Ohjelmista tulostettava teksti voidaan ohjata kirjoittimelle seuraavasti:

```
PRINT #8,"Tämä lähtee kirjoittimelle"
```

Monet kirjoittimet toimivat siten, että ne eivät kirjoita riviä täyteen, vaan alkavat tulostaa sanaa seuraavalle riville, jos se on liian pitkä mahtuakseen rivin loppuun. Tämä seikka on syytä varmistaa kirjoittimen käsikirjasta. Tulostettavan rivin leveys on määriteltävissä AMSTRADin BASICin WIDTH-käskyllä. Oletusarvona käytetään kirjoittimen tulostusrivin leveyttä 132, mutta se voidaan asettaa tarpeen vaatiessa toiseksikin esim. käskyllä WIDTH 80.

Jos arvo asetetaan 255:ksi, ei AMSTARDin BASIC katkaise rivejä ennen niiden täyttymistä ja toiminta riippuu aivan kokonaan kirjoittimen toiminnasta. AMSTRADin BASICissa on myös kirjoittimen tilaa koskeva laskuri, ja kirjoittimen tila voidaan saada selville POS-toiminnolla:

```
IF POS(#8) > 50 THEN GOTO 100
```

CPC 464 muodostaa rivin lopussa rivinvaihtokomennon CHR$(10) ja vaununpalautuskomennon CHR$(13). Kirjoittimissa on yleensä kytkimet, joilla valitaan sopiva tulostettavan tiedon vastaanottotapa. Saat selville oman kirjoittimesi oletusarvot heti kun yrität tulostaa kirjoittimella jotakin.

7.3 GRAFIIKKAKIRJOITIN

Kirjoittimen mukana toimitetussa käsikirjassa on määritelty ohjauskoodit, jotka ovat yleensä muotoa:

```
PRINT CHR$(n)
```

Joidenkin kirjoittimien merkkivalikoimaan voi kuulua merkkejä, jotka ovat samanlaisia kuin useimmat liitteessä III luetellut AMSTRADin grafiikkamerkitkin. On kuitenkin varsin epätodennäköistä, että merkkien numerot vastaavat täysin toisiaan, joten saatat ehkä tarvita oman muuntotaulukon, jolla saat kirjoittimen tulostamaan haluamasi merkit.

Vaikka CPC 464:n kirjoitinliitäntä on ensisijassa tarkoitettu huokeille pistematriisikirjoittimille, sitä voidaan kuitenkin käyttää myös sopivalla liitännällä varustettujen kirjasinkiekkokirjoittimienkin kanssa sekä eräiden piirtureiden, ja moniväristen mustesuihkupiirtureiden kanssa. Liitännän sopivuuden ratkaisee se, onko kirjoittimessa tai piirturissa käytetty vakiomallista rinnakkaisliitäntää.

LUKU 8: LYHYT KATSAUS AMSTRADIN BASIC-KIELEEN

Tässä luvussa on lueteltu kaikki AMSTRADin BASIC-kielen avainsanat niiden käyttöön liittyvine muine avainsanoineen. Avainsanat on esitetty täsmälleen siinä muodossa jossa niitä on käytettävä. Lisäksi avainsanojen käyttöä on kuvattu lukuisilla esimerkeillä.

Tässä luvussa käsiteltäviä asioita ovat:

- Avainsanojen merkintätapa
- Erikoismerkit ja niiden merkitys
- Kaikki AMSTRADin BASIC-avainsanat aakkosjärjestyksessä

Tässä luvussa käsitellään lyhyesti CPC 464:n ROM-muistissa olevien BASIC-avainsanojen toimintaa. CPC 464:n ominaisuuksiin kuuluu laaja valikoima vakiintuneita BASIC-toimintoja. Niiden lisäksi käytettävissä on joukko CPC 464:lle tyypillisiä toimintoja.

8.1 MERKINTÄTAPA

Erikoismerkit

& tai &H	Heksadesimaalimuotoisen vakion etuliite
&X	Binäärimuotoisen vakion etuliite
:	Erottaa kaksi samalle riville näppäiltyä lauseketta
#	Ohjaa tulostuksen halutulle tulostusreitille (esim. tiettyyn ikkunaan tai kirjoittimelle)

Datatyypit

Merkkijonojen pituus voi olla 0 - 255 merkkiä, ja niitä kutsutaan 'merkkijonolausekkeiksi'. Merkkijonot liitetään toisiinsa käyttämällä "liitos-operaattoria" niin kauan kun tulokseksi saatavan 'merkkijonolausekkeen' pituus on alle 255 merkkiä.

Numerotiedot voivat olla joko kokonaislukuja tai reaalilukuja. CPC 464:n kokonaislukualue on -32768...+32767. Reaalilukualue on ±1,7E+38, jolla tarkkuus on hieman yli 9 numeroa. Pienin nollan alapuolella oleva arvo on noin 2,9E-39.

Kokonaislukuja merkitään prosenttimerkillä **%** , reaalilukuja merkitään huutomerkillä **!** ja merkkijonoja merkitään dollarinmerkillä **$** .

'Numerolauseke' on mikä hyvänsä lauseke, jonka tulokseksi saadaan jokin numeroarvo. Siispä se voi olla pelkkiä numeroita, numeromuuttuja tai joukko muuttujilla operoitavia lukuja; siis lähes melkein mitä hyvänsä, mikä ei ole 'merkkijonolauseke'.

'Reittilauseke' viittaa 'numerolausekkeeseen' jolla määritellään haluttu "tulostusreitti". Se voi olla kuvaruutu tai osa siitä (ikkuna), kirjoitin tai kasettinauhuri.

Ruutuun ilmestyvä ilmoitus **improper argument** (sopimaton argumentti) tarkoittaa sitä, että 'numerolausekkeesta' on saatu tulokseksi arvo, joka on sen alueen ulkopuolella, joka on määritelty sallituksi kyseisessä tilanteessa. Toinen syy ilmoitukseen voi olla se, että käskyssä käytetty parametri on tavalla tai toisella epäkelpo, eikä BASIC ole hyväksynyt sitä.

AVAINSANAT

AMSTRADin BASIC-kielen avainsanat on lueteltu seuraavassa muodossa:

AVAINSANA
Muoto/toiminta
Esimerkki
Kuvaus
Aiheeseen liittyvät AVAINSANAT

Tärkeää:

Avainsanat ovat joko KÄSKYJÄ (t. KOMENTOJA) eli operaatioita jotka suoritetaan välittömästi, tai FUNKTIOITA eli operaatioita joita käytetään argumentteina erilaisissa lausekkeissa.

Sulkumerkit

Tavallisia sulkumerkkejä ( ) käytetään monen käskyn ja funktion osana. Ja koska ne kuuluvat osana useisiin käskyihin ja funktioihin, on ne myös näppäiltävä näppäimistöltä. Muunlaisia sulkumerkkejä ei näppäillä mukaan käskyyn, vaan niitä käytetään käskyn rakennetta kuvaavissa selvityksissä. Hakasulkeita [ ] käytetään erilaisten valinnaisten (= ei pakollisten) käskyn osien kuvaamiseen.

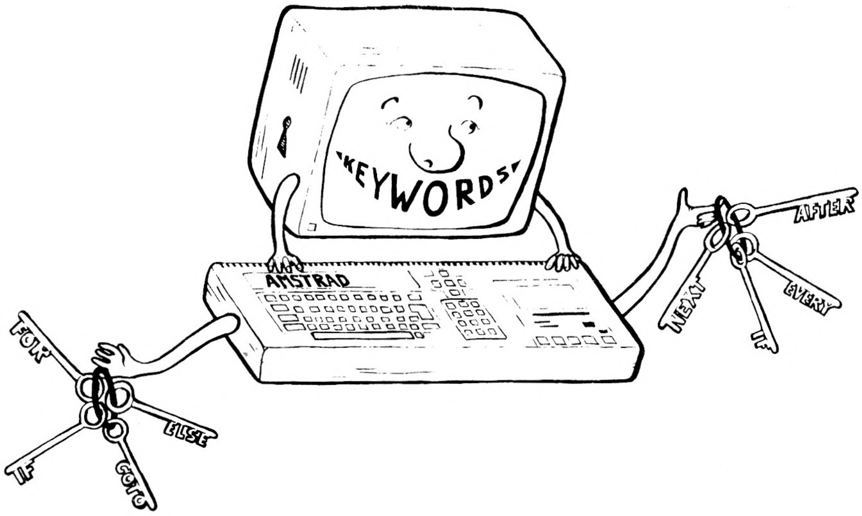
Lainausmerkit

Tavallisia lainausmerkkejä " " käytetään osana BASIC-ohjelmia esim. PRINT-käskyjen yhteydessä. Muunlaisia lainausmerkkejä ei näppäillä mukaan käskyihin tai funktioihin.
Merkkejä ' ' käytetään erilaisten lausekkeiden nimien yhteydessä. Kuhunkin käskyyn tai funktioon liittyviä lausekkeita kuvataan kyseisen käskyn tai funktion yhteydessä.
(Merkkien ' ' tilalla on käytetty käsikirjan alkuosassa merkkejä < >).

Avainsanojen syöttäminen näppäimistöltä

AMSTRADin BASIC muuttaa pienillä kirjaimilla näppäillyt avainsanat suurilla kirjaimilla tulostuviksi avainsanoiksi, kun ohjelmalistaus pyydetään LIST-käskyllä. Seuraavissa esimerkeissä avainsanat on kirjoitettu suurilla kirjaimilla, koska ne esiintyvät siinä muodossa listattaessa. Niinpä avainsanat kannattaakin näppäillä pienillä kirjaimilla, jolloin näet helpommin mahdolliset kirjoitusvirheet pyytäessäsi ohjelmalistauksen LIST-käskyllä, koska väärin näppäillyt avainsanat näkyvät edelleen pienillä kirjaimilla kirjoitettuina.

Avainsanat on erotettava erotusmerkeillä, koska AMSTRADin BASIC sallii avainsanojen käytön muuttujien nimissä. Esim. end2 ja LISTCODE ovat hyväksyttäviä muuttujia AMSTRADin BASICissa.



ABS

ABS(´numerolauseke´)

```
PRINT ABS(-67.98)
 67.98
```

FUNKTIO: Antaa suluissa määritellyn numerolausekkeen itseisarvon. Tämä tarkoittaa käytännössä lähinnä sitä, että negatiivisten lukujen arvo saadaan positiivisena.

Aiheeseen liittyvät avainsanat: SGN

AFTER

AFTER ´kokonaislukulauseke´[,´kokonaislukulauseke´] GOSUB ´rivinumero´

```
AFTER 200,2 GOSUB 320
```

KÄSKY: Kutsuu aliohjelmaa sen jälkeen kun määritelty aika on kulunut. Ensimmäinen ´kokonaislukulauseke´ määrittelee viiveen pituuden 0,02 sekunnin pituisissa yksiköissä. Toinen ´kokonaislukulauseke´ (väliltä 0 - 3) määrittelee yhden neljästä käytettävissä olevasta viiveajastimesta.

Aiheeseen liittyvät avainsanat: EVERY, REMAIN

ASC

ASC(´merkkijonolauseke´)

```
PRINT ASC("X")
 88
```

FUNKTIO: Antaa merkkijonon ensimmäisen merkin numeroarvon niin kauan kun käytetään ASCII-merkkejä.

Aiheeseen liittyvät avainsanat: CHR$

ATN

ATN('numerolauseke')

```
PRINT ATN(1)
 0.785398163
```

FUNKTIO: Laskee määritellyn arvon arkustangentin (arc tan) (pakottaen 'numerolausekkeen' reaaliluvuksi ilmaistuna radiaaneissa välillä PI/2 - PI/2).

Aiheeseen liittyvät avainsanat: SIN, COS, TAN, DEG, RAD

AUTO

AUTO ['rivinumero'][,'inkrementti']

```
AUTO 100,50
```

KÄSKY: Muodostaa automaattisesti rivinumeroita. Kohdassa 'rivinumero' määritellään ensimmäinen muodostettava rivinumero mikäli käskyä halutaan käyttää jo olemassa olevan ohjelman perään lisättävässä ohjelmassa. Kohdan 'inkrementti' arvo määrää rivinumeroiden välin suuruuden eli askelvälin. CPC 464 käyttää molemmissa oletusarvoa 10 ellei sitä erikseen ole määritelty.

AMSTRADin BASIC lisää tähtimerkin * AUTO-käskyllä muodostetun rivinumeron jälkeen varoitukseksi, jos jokin olemassa olevan ohjelman rivi on vaarassa tulla korvatuksi tällä uudella rivillä. Jos haluat säilyttää vanhan rivin sellaisenaan, paina [ENTER]-näppäintä. Paina lopuksi [ESC]-näppäintä kun et enää tarvitse automaattista rivinumerointia.

BIN$

BIN$('etumerkitön kokonaislukulauseke'[,'kokonaislukulauseke'])

```
PRINT BIN$(64,8)
01000000
```

FUNKTIO: Muuttaa kohdassa 'etumerkitön kokonaislukulauseke' määritellyn luvun binäärimuotoon käyttäen kohdassa 'kokonaislukulauseke' annetun määrän numeroita. Binääriluvun alkuosa täytetään tarvittaessa etunollilla.

Aiheeseen liittyvät avainsanat: HEX$, STR$

BORDER

BORDER 'väri'[,'väri']

```
BORDER 3,2
```

KÄSKY: Käytetään kuvaruudun reuna-alueen värin vaihtamiseen. Reuna-alueen väri vuorottelee kahden värin välillä, mikäli BORDER-käskyllä määritellään kaksi pilkulla erotettua väriä. Värien vuorottelunopeus määräytyy tällöin SPEED INK-käskyllä määritellyllä nopeudella (ei pakko määritellä). Reuna-alueen värien värinumerot voivat olla väliltä 0 - 26.

Aiheeseen liittyvät avainsanat: SPEED INK

CALL

CALL ´osoitelauseke´[,´parametriluettelo´]

CALL &BD19

KÄSKY: Tämän käskyn avulla voidaan BASICista kutsua ulkoisesti mudostettua aliohjelmaa. Tätä käskyä on syytä käyttää varoen, eikä sen kanssa kannata alkaa suorittamaan kokeiluja, ellei tietokoneista ja niiden ohjelmoinnista ole riittävästi kokemusta. Yllä olevassa esimerkissä esitetty CALL-käsky on suhteellisen harmiton, koska se odottaa seuraavaa kuvan kentän vaihtoa. Niinpä käsky sopiikin oikein hyvin käytettäväksi erilaisten kuvioiden liikkeiden pehmentämiseen kuvaruudulla.

Aiheeseen liittyvät avainsanat: UNT

CAT

CAT

CAT

KÄSKY: Tällä käskyllä saadaan selville kasetilla olevien tiedostojen tai ohjelmien nimet, jotka tulostuvat kuvaruutuun. CAT-käsky ei vaikuta parhaillaan ohjelmamuistissa olevaan ohjelmaan, joten sitä voidaankin käyttää SAVE-käskyllä juuri kasetille tallennetun ohjelman oikeellisuuden varmistamiseen (verify). CAT-käskyn annettuasi tietokone pyytää sinua painamaan kasettinauhurin PLAY-painiketta, jonka jälkeen ruutuun ilmestyy kasetilta löytyvien tiedostojen nimet seuraavassa muodossa:

OHJELMAN NIMI   Block Nro Lippu Ok

Lippu ilmoittaa nauhalle tallennetun tiedoston tyypin seuraavasti:
- $ BASIC-ohjelmatiedosto
- % suojattu BASIC-tiedosto
- * ASCII-tekstitiedosto
- & binäärimuotoinen tiedosto

Lippumerkkinä voi olla jokin muukin, jos ohjelmaa ei ole alunperin tehty BASICilla. Rivin lopussa oleva ilmoitus Ok tarkoittaa sitä, että tiedosto on virheetön ja se olisi latautunut muistiin LOAD-käskyllä.

"Block" tarkoittaa tiedoston lohkoa ja "Nro" sen numeroa.

Aiheeseen liittyvät avainsanat: LOAD, RUN, SAVE

CHAIN
CHAIN MERGE

CHAIN ´tiedoston nimi´[,´rivinumerolauseke´]
CHAIN MERGE ´tiedoston nimi´[,´rivinumerolauseke´][,DELETE ´rivinumeroalue´]

CHAIN "TESTI", 350

KÄSKY: CHAIN-käsky lataa kohdassa ´tiedoston nimi´ määritellyn ohjelman kasetilta tietokoneen muistiin, jolloin muistissa mahdollisesti ollut entinen ohjelma tuhoutuu. CHAIN MERGE-käsky limittää kohdassa ´tiedoston nimi´ määritellyn ohjelman kasetilta sillä hetkellä ohjelmamuistissa olevaan ohjelmaan. Se siis lisää tiedoston sisällön muistissa olevaan ohjelmaan. Kohdassa ´rivinumerolauseke´ määritellään se rivinumero, josta ohjelman suoritus käynnistyy heti kun uusi ohjelma on latautunut muistiin. Jos kohdassa ´rivinumerolauseke´ ei anneta mitään määritettä, BASIC olettaa, että ohjelman suoritus käynnistetään pienimmästä rivinumerosta.

Jos tiedoston nimeä ei määritellä, BASIC yrittää limittää ensimmäisen virheettömän nauhalta löytämänsä tiedoston. Jos tiedoston nimen ensimmäinen merkki on **!** , niin BASIC poistaa sen tiedoston nimestä ja jättää tulostamatta ruutuun tavanomaisia kehoitusviestejä.

CHAIN MERGE-käsky säilyttää kaikki senhetkisten muuttujien arvot, mutta käyttäjän toiminnot ja avoimet tiedostot hylätään. ON ERROR GOTO kytketään pois toiminnasta, RESTORE toteutetaan ja DEFINT-, DEFREAL- & DEFSTR-asetukset palautetaan alkuarvoihinsa, ja kaikki aktiiviset FOR-, WHILE- ja GOSUB-käskyt unohdetaan. Suojattuja tiedostoja ei voi limittää.

Aiheeseen liittyvät avainsanat: LOAD, MERGE

CHR$

CHR$(´kokonaislukulauseke´)

```
PRINT CHR$(100)
 d
```

FUNKTIO: Muuttaa numeroarvon vastaavaksi merkiksi (käyttäen liitteessä III lueteltua AMSTRAD CPC 464:n merkkivalikoimaa).

Aiheeseen liittyvät avainsanat: ASC, LEFT$, RIGHT$, MID$, STR$

CINT

CINT(´numerolauseke´)

```
10 n = 578.76543
20 PRINT CINT(n)
RUN
 579
```

FUNKTIO: Muuttaa ´numerolausekkeen´ kokonaisluvuksi välillä -32768...+32767.

Aiheeseen liittyvät avainsanat: CREAL, INT, FIX, ROUND, UNT

CLEAR

CLEAR

CLEAR

KÄSKY: Nollaa kaikki muuttujat ja tiedostot.

CLG

CLG[´peittomuste´]

CLG

KÄSKY: Tyhjentää grafiikkanäytön. Väri on mahdollista valita kohdassa ´peittomuste´.

Aiheeseen liittyvät avainsanat: CLS, ORIGIN

CLOSEIN

CLOSEIN

CLOSEIN

KÄSKY: Sulkee kasetin syöttötiedoston (input file). Käskyjen NEW ja CHAIN MERGE kaltaiset käskyt hylkäävät kaikki avoimet tiedostot.

Aiheeseen liittyvät avainsanat: OPENIN, CLOSEOUT

CLOSEOUT

CLOSEOUT

CLOSEOUT

KÄSKY: Sulkee kasetin tulostustiedoston (output file).

Aiheeseen liittyvät avainsanat: OPENOUT, CLOSEIN

CLS

CLS [#´reittilauseke´]

CLS #0

KÄSKY: Käsky tyhjentää määritellyn ikkunan, jolloin ikkunan väriksi tulee paperin musteen väri.

CONT

CONT

CONT

KÄSKY: Käsky jatkaa ohjelman suoritusta ***Break*** -ilmoituksen, STOP- tai END-käskyn jälkeen kunhan ohjelmaa ei vain ole muutettu. Näppäimistöltä voidaan kuitenkin antaa suoria käskyjä.

COS

COS(´numerolauseke´)

?COS(34)	tai	deg:?cos(34)
-0.848570274		0.829037573

FUNKTIO: Laskee suluissa määritellyn numerolausekkeen kosinin. Funktion mittayksikkönä käytetään oletusarvona radiaaneja, ellei sitä ole erikseen määritelty asteiksi DEG-käskyllä. Yllä olevassa esimerkissä on käytetty kysymysmerkkiä **?** PRINT-käskyn lyhenteenä, ja avainsanat on näppäilty pienillä kirjaimilla. Kaikki avainsanathan voidaan näppäillä myös pienillä kirjaimilla.

Aiheeseen liittyvät avainsanat: SIN, TAN, ATN, DEG, RAD

CREAL

CREAL(´numerolauseke´)

```
5 DEFINT n
10 n=75.765
20 d=n/34.6
30 PRINT d
40 PRINT CREAL(n)
50 PRINT n/55.4
run
2.19653179
76
1.37184116
Ready
```

FUNKTIO: Muuttaa ´numerolausekkeen´ reaaliluvuksi eli luvuksi, jossa on kokonaisosa ja desimaaliosa (vastakohtana kokonaisluku, jossa ei ole desimaaliosaa).

Aiheeseen liittyvät avainsanat: CINT

DATA

DATA ´joukko vakioita´

```
10 REM Nimiluettelo
20 DIM Etunimi$(5)
30 DIM Sukunimi$(5)
40 FOR n=1 TO 5
50 READ Etunimi$(n)
60 READ Sukunimi$(n)
65 PRINT Etunimi$(n);" "Sukunimi$(n)
70 DATA Pekka,Ponteva,Liisa,Lahjakas,Tai
na,Topakka,Vieno,Vauhdikas,Tauno,Taitava
90 NEXT
```

KÄSKY: DATA-käskyllä määritellään ohjelmassa käytettäviä DATA-arvoja. Tämä on eräs kaikkein yleisimmistä BASIC-ohjelman piirteistä, eli vakiotiedot sijoitetaan DATA-lausekkeisiin, joista ne saadaan ohjelmaan aina tarvittaessa. DATA-tyypin on oltava vastaavaa muotoa kuin sitä kutsuvan muuttujankin. DATA-lauseke voi olla missä kohdassa ohjelmaa hyvänsä.

Aiheeseen liittyvät avainsanat: READ, RESTORE

DEF FN

DEF FN´nimi´[(´muodolliset parametrit´)]=´yleislauseke´

```
10 DEF FNkorko(rahat)=1.14*rahat
20 INPUT "Paljonko on rahaa";rahat
30 PRINT "Rahat korkoineen vuoden kulutt
ua";FNkorko(rahat)
```

KÄSKY: Käyttäjä voi määritellä BASIC-ohjelmalla yksinkertaisia funktioita, joista saadaan vastauksiksi numeroarvoja. Funktio määritellään käskyllä DEF FN, ja sillä luodaan ohjelmakohtainen funktio, joka toimii ohjelman puitteissa samalla tavalla kuin esim. BASICin kiinteisiin funktioihin kuuluva COS-funktiokin.

Määriteltyä funktiota voidaan kutsua koko ohjelman puitteissa. Muuttujien on oltava vastaavaa tyyppiä, ja DEF FN-käsky on kirjoitettava ohjelmaan suoritussilmukan ulkopuolelle.

DEFINT
DEFSTR
DEFREAL

DEFtyyppi ´kirjainalue (tai -alueet)´

```
DEFINT I-N
DEFSTR A,W-Z
DEFREAL
```

KÄSKYJÄ: Näillä käskyillä määritellään muuttujien oletustyypit, jossa "tyyppi" on kokonaisluku (INT), merkkijono (STR) tai reaaliluku (REAL). Muuttuja määräytyy sen nimen ensimmäisen kirjaimen mukaan, joka voi olla joko suuri tai pieni kirjain.

Aiheeseen liittyvät avainsanat: LOAD, RUN, CHAIN, NEW, CLEAR

DEG

DEG

DEG

KÄSKY: Tällä käskyllä valitaan toimintatila, jossa mittayksikkönä käytetään asteita. Tietokone valitsee mittayksiköksi oletusarvona radiaanit sinin (SIN) ja kosinin (COS) kaltaisille funktioille. DEG-käsky asettaa käyttöön siis asteet, kunnes toimintatila vaihdetaan CLEAR- tai RAD-käskyllä, tai kun ohjelmamuistiin ladataan uusi ohjelma.

Aiheeseen liittyvät avainsanat: RAD

DELETE

DELETE ´rivinumeroalue´

DELETE 100-200 tai DELETE -300 tai DELETE 150-

KÄSKY: DELETE-käskyllä voidaan poistaa kohdassa ´rivinumeroalue´ määritelty osa parhaillaan ohjelmamuistissa olevasta ohjelmasta. DELETE-käskyllä poistettuja rivejä ei enää saa palautetuksi ohjelmaan, jos DELETE-käskyä käytetään vahingossa. Tästä syystä käskyä on käytettävä varsin harkitusti, ja määritelty rivinumeroalue on syytä tarkistaa, ettei ohjelmasta tuhota vääriä rivejä.

Aiheeseen liittyvät avainsanat: NEW

DI

DI

```
10 CLS
20 TAG
30 EVERY 10 GOSUB 100
40 X1=RND*320:X2=RND*320
50 Y=200+RND*200
60 FOR X=320-X1 TO 320+X2 STEP 2
70 DI:PLOT 320,0,1:MOVE X-2,Y:PRINT " ";
:MOVE X,Y:PRINT "#";:EI
80 NEXT
90 GOTO 40
100 MOVE 320,0
110 DRAW X+8,Y-16,3
120 RETURN
```

KÄSKY: DI-käsky estää keskeytykset (muut kuin ***Break*** -keskeytykset), kunnes keskeytykset on jälleen sallittu. Keskeytysten sallinta voidaan määritellä erikseen EI-käskyllä, mutta keskeytykset ovat myös sallittuja sen jälkeen kun keskeytyksen aiheuttanut GOSUB-aliohjelma päättyy RETURN-käskyyn.

DI-käskyä käytetään silloin, kun jokin ohjelman osa halutaan suorittaa kirjaimellisesti keskeytyksettä esimerkiksi silloin, kun kaksi saman ohjelman osaa kilpailevat käytettävissä olevista resursseista. Yllä olevassa esimerkkiohjelmassa pääohjelma ja keskeyttävä aliohjelma kilpailevat grafiikkanäytön käytöstä.

Aiheeseen liittyvät avainsanat: EI

DIM

DIM ´indeksoitujen muuttujien luettelo´

```
10 CLS:PRINT "Anna 5 nimeä....":PRINT
20 DIM B$(5)
30 FOR N=1 TO 5
40 PRINT "Nimi"N;
50 INPUT B$(N)
60 NEXT
70 FOR N=1 TO 5
80 PRINT "Teit viisaasti ";B$(N);" kun o
stit CPC 464:n"
90 NEXT
```

KÄSKY: DIM-käskyllä varataan tilaa taulukoille ja määritellään indeksien maksimiarvot. BASIC-ohjelmassa täytyy erikseen määritellä, kuinka paljon taulukolle tarvitaan tilaa. Muussa tapauksessa käytetään oletusarvoa 10. Kun taulukon koko on kerran määritelty, sitä ei voida enää muuttaa, tai siitä on seurauksena virheilmoitus.

Indeksoitu muuttuja on sellainen, jossa sama muuttujan nimi voi saada useita eri arvoja sen mukaan, kuinka suureksi taulukko on mitoitettu DIM-käskyllä. Taulukon koko määräytyy DIM-käskyn ´indeksoitujen muuttujien luettelo´-osassa olevien kokonaislukujen mukaan. Sulkeissa voi siis olla useampia pilkulla erotettuja kokonaislukuja, jolloin taulukko on moniulotteinen. Luettelon ensimmäistä lukua voitaisiin verrata vaikkapa monikerroksisen paikoitustalon kerrosten lukumäärään, ja luettelon seuraavat numerot voisivat vastata esim. kussakin kerroksessa olevien autopaikkojen määrää, jne. Taulukoiden täydellinen ymmärtäminen on varsin keskeinen asia edistyneessä BASIC-ohjelmoinnissa. Taulukon kokoa rajoittaa ainoastaan käytettävissä oleva muistikapasiteetti ja ohjelmoijan kyky pysyä kärryillä taulukon kuhunkin ulottuvuuteen kuuluvista arvoista.

Aiheeseen liittyvät avainsanat: ERASE

DRAW

DRAW ´x-koordinaatti´,´y-koordinaatti´[,´peittomuste´]

DRAW 200,200,13

KÄSKY: Piirtää viivan kuvaruutuun senhetkisestä grafiikkakohdistimen paikasta x- ja y-koordinaateilla määriteltyyn paikkaan. Koordinaattipaikat eivät muutu, vaikka näyttömuotoa (MODE) vaihdetaan. Aiheesta on lisää esimerkkejä luvussa 5.

Aiheeseen liittyvät avainsanat: DRAWR, PLOT, PLOTR, MOVE, MOVER, TEST, TESTR, XPOS, YPOS, ORIGIN

DRAWR

DRAWR ´x-siirtymä´,´y-siirtymä´[,´peittomuste´]

DRAWR 200,200,13

KÄSKY: Piirtää viivan kuvaruutuun senhetkisestä grafiikkakohdistimen paikasta paikkaan, joka määritellään x-suunnan ja y-suunnan siirtymillä.

Aiheeseen liittyvät avainsanat: DRAW, PLOT, MOVE, MOVER, TEST, TESTR, XPOS, YPOS, ORIGIN

EDIT

EDIT ´rivinumero´

EDIT 110

KÄSKY: Tällä käskyllä kutsutaan tietty rivinumero esille korjattavaksi. Katso lukua 1.4.

Aiheeseen liittyvät avainsanat: LIST

EI

EI

EI

KÄSKY: Tällä käskyllä sallitaan DI-käskyllä estetyt keskeytykset.

Aiheeseen liittyvät avainsanat: DI

END

END

END

KÄSKY: Tämä käsky päättää ohjelman. END sisältyy epäsuorasti AMSTRADin BASICiin ohjelman kulkiessa viimeisen käskyrivin läpi. END-käsky sulkee kaikki kasettitiedostot ja palauttaa toiminnan suoraan toimintatilaan, jolloin käskyjä voidaan syöttää suoraan näppäimistöltä. SOUND-käskyllä määritellyissä äänijonoissa olevat äänet jatkavat soimistaan, kunnes jonot ovat tyhjät.

Aiheeseen liittyvät avainsanat: STOP

ENT

ENT ´vaipan numero´[,´vaipan osat´]

```
10 ENT 1,100,2,20
20 SOUND 1,100,1000,4,0,1
```

KÄSKY: Kun jotakin ääntä muodostetaan, voidaan sen sävelkorkeutta muuttaa. Tapa jolla sävelkorkeutta muutetaan, määritellään sävelvaipalla. SOUND-käskyä ja sen käyttöä on kuvattu tarkemmin luvussa 6 ja liitteessä VII.

ENT-käskyn osa ´vaipan numero´ on kokonaisluku, jonka itseisarvon on oltava väliltä 1 - 15. Se määrittelee sävelvaipan numeron. Jos ´vaipan numero´ on negatiivinen, vaippa toistetaan.

Määriteltäviä 'vaipan osia' voi olla enintään viisi, ja kukin niistä voi olla muodoltaan jompaa kumpaa alla olevaa tyyppiä:

- 'portaiden määrä','portaan koko','portaan kesto'
- ='säveljakso','portaan kesto'

Ensimmäinen muoto määrittelee muutoksen suuruuden verrattuna senhetkiseen säveljaksoasetukseen. Toinen muoto määrittelee säveljaksolle absoluuttisen arvon. Kohdassa 'portaiden määrä' määritellään kyseisessä vaipan osassa olevien portaiden määrä, joka voi olla jokin kokonaisluku väliltä 0...239.

Kohdassa 'portaan koko' määritellään, kuinka paljon säveljaksoa muutetaan kunkin vaippaan kuuluvan portaan kohdalla. Se voi olla jokin kokonaisluku väliltä -128...+127.

Kohdassa 'portaan kesto' määritellään kahden portaan välinen aika eli portaan kesto. Se voi olla kokonaisluku väliltä 0...255 (jossa CPC 464 käsittelee lukua 0 lukuna 265). Portaan keston yksikkönä käytetään 1/100 sekuntia (0,01 s).

Kohdassa 'säveljakso' määritellään jakson uusi asetus. Se voi olla jokin kokonaisluku väliltä 0...4095.

Alkusäveljakso asetetaan SOUND-käskyssä, ja siinä voidaan määritellä yksi viidestätoista sävelvaipasta. Sävelkorkeus pysyy vakiona koko äänen ajan, ellei mitään sävelvaippaa määritellä tai jos määritellään sellainen vaippa, jota ei ole asetettu.

Sävelvaipalla ei ole mitään vaikutusta äänen kestoon. Jos sävelvaippaan kuuluvia portaita on jäljellä äänen loppuessa, ne yksinkertaisesti hylätään.

Toistuva sävelvaippa käynnistetään uudelleen aina sen päätyttyä, kunnes ääni lakkaa.

Tietokone laskee sävelvaippaan kuuluvien lausekkeiden arvon käskyä suoritta-essaan ja tallentaa tulokset myöhempää käyttöä varten. Sävelvaipan käyttö ei aiheuta käskyn uudelleen suorittamista.

Joka kerta kun jokin tietty sävelvaippa asetetaan, häviää sen aikaisempi arvo muistista. Jos vaippaa muutetaan silloin kun sitä käyttävä ääni on parhaillaan soimassa tai odottaa vuoroaan jonossa, on sen vaikutus ääneen määrittelemätön (saattaa olla kiinnostavakin).

Kaikki aikaisemmat asetukset peruuntuvat määriteltäessä vaippa, johon ei kuulu yhtään osaa. Jos kyseistä vaippaa yritetään tämän jälkeen käyttää, ei sitä huomioida vaan sen tilalla käytetään oletusarvoja.

Aiheeseen liittyvät avainsanat: ENV, SOUND

ENV

ENV ´vaipan numero´[,´vaipan osat´]

```
10 ENV 1,100,2,20
20 SOUND 1,100,1000,4,1
```

KÄSKY: Ääntä muodostettaessa on mahdollista muuttaa sen voimakkuutta ENV-käskyllä. Voimakkuusvaipan osia ovat ´vaipan numero´, ´portaiden määrä´, ´portaan koko´, ´portaan kesto´.

Portaiden määrä voi olla jokin kokonaisluku väliltä 0...127, portaan koko voi olla jokin kokonaisluku väliltä -128...+127 ja portaan kesto voi olla jokin kokonaisluku väliltä 1...256. Portaan keston mittayksikkönä käytetään 1/100 sekuntia (0,01 s).

Vaipan numero voi olla puolestaan jokin kokonaisluku väliltä 1 - 15, ja sillä määritellään asetettava voimakkuusvaippa.

Kussakin voimakkuusvaipassa voi olla enintään viisi ´vaipan osaa´, ja kukin niistä voi olla muodoltaan jompaa kumpaa alla olevaa tyyppiä:

- ´portaiden määrä´,´portaan koko´,´portaan kesto´
- =´valmisvaippa´,´vaippajakso´

Ensimmäinen muoto määrittelee vaipan osan ohjelmallisesti, jossa eri määritteet ovat:

- ´portaiden määrä´ ilmoittaa kyseisessä vaipan osassa olevien portaiden lukumäärän. Se voi olla jokin kokonaisluku väliltä 0...127.

- ´portaan koko´ ilmoittaa, kuinka paljon portaan amplitudia muutetaan kunkin portaan kohdalla. Se voi olla jokin kokonaisluku väliltä -128...+127. Jos portaiden määrä on 0, niin arvo, johon amplitudi asetetaan, on absoluuttinen arvo.

- ´portaan kesto´ määrittelee kahden portaan välisen ajan eli portaan keston. Se voi olla jokin kokonaisluku väliltä 0...255 (jossa CPC 464 käsittelee lukua 0 lukuna 265). Yksikkönä käytetään 1/100 sekuntia (0,01 s).

Toinen muoto määrittelee vaipan osan, joka suoritetaan ilman ohjelmaohjausta käyttäen laitteiston omia äänenmuodostustoimintoja. Vaipan osia ovat:

- ´valmisvaippa´. Tämä on arvo, joka asetetaan vaipanmuotorekisteriin (rekisteri 15; oktaalimuoto).

- ´vaippajakso´. Tämä on arvo, joka asetetaan vaippajaksorekistereihin (rekisterit 13 ja 14; oktaalimuoto).

Laitteiston omiin "valmisvaippa-asetuksiin" ei liity portaan kestoa määrittelevää osaa, joten seuraava vaipan osa suoritetaan välittömästi edellisen jälkeen. Tästä syystä kannattaa seuraavalle portaalle määritellä riittävä pituus. Jos seuraavaa porrasta ei ole määritelty, käytetään portaan keston oletusarvona kahta sekuntia.

Alkuäänenvoimakkuus asetetaan SOUND-käskyssä, ja siinä voidaan määritellä yksi viidestätoista voimakkuusvaipasta. Voimakkuus pysyy vakiona koko äänen ajan, ellei mitään vaippaa määritellä tai jos määritellään sellainen vaippa, jota ei ole asetettu.

Jos ´portaan koko´ määritellään nollaksi silloin kun portaiden määrä on jokin nollasta poikkeava luku, käytetään senhetkistä voimakkuusasetusta.

Tietokone laskee voimakkuusvaipan lausekkeiden arvot suorittaessaan käskyn ja tallentaa tulokset myöhempää käyttöä varten. Voimakkuusvaipan käyttäminen ei aiheuta käskyn uudelleensuorittamista.

Joka kerta kun jokin tietty voimakkuusvaippa asetetaan, häviää sen aikaisempi arvo muistista. Jos vaippaa muutetaan silloin kun sitä käyttävä ääni on parhaillaan soimassa tai odottaa vuoroaan jonossa, on sen vaikutus ääneen määrittelemätön (saattaa olla kiinnostavakin).

Kaikki aikaisemmat asetukset peruuntuvat määriteltäessä vaippa, johon ei kuulu yhtään osaa. Jos kyseistä vaippaa yritetään tämän jälkeen käyttää, ei sitä huomioida vaan sen tilalla käytetään oletusarvoja.

Aiheeseen liittyvät avainsanat: ENT, SOUND

EOF

EOF

PRINT EOF
-1

FUNKTIO: Tarkistaa, onko kasettisyöttö tiedoston lopun kohdalla. Vastaus on -1 (tosi) tiedoston lopun kohdalla ja muussa tapauksessa 0 (epätosi).

Aiheeseen liittyvät avainsanat: OPENIN

ERASE

ERASE ´muuttujanimien luettelo´

ERASE A,B$

KÄSKY: Kun DIM-käskyllä määriteltyä taulukkoa ei enää tarvita, se voidaan tyhjentää ERASE-käskyllä. Tällä tavalla vapautuu muistikapasiteettia muuhun käyttöön.

Aiheeseen liittyvät avainsanat: DIM

ERR
ERL

ERR
ERL

```
10 CLS
20 ON ERROR GOTO 1000
30 DATA VIENO,KAINO,EINO,AINO,UUNO
40 READ A$
50 PRINT A$
60 GOTO 40
70 REM virheenpaljastus alkaa tästä
1000 IF ERL<70 AND ERL>20 THEN PRINT "Ri
veiltä 30 - 60"
1010 IF ERR=4 THEN PRINT "löytyi DATA EX
HAUSTED-virhe!"
1020 END
```

MUUTTUJIA: Näitä muuttujia käytetään virheidenkäsittelyaliohjelmissa paljastamaan virheen numero (ERR) ja rivi (ERL) jolla virhe on esiintynyt. Katso liitteessä VIII olevia virheilmoitusten selityksiä.

Aiheeseen liittyvät avainsanat: ON ERROR, ERROR

ERROR

ERROR ´kokonaislukulauseke´

ERROR 17

KÄSKY: Tämä käsky aiheuttaa kokonaisluvun määräämää virhenumeroa vastaavan toiminnon. Virhe voi olla jo aikaisemmin BASICin käyttämä ja tunnistama (katso liitettä VIII), jolloin toiminto on sama kuin mikä tapahtuisi silloin, jos BASIC olisi havainnut saman virheen. Muita kuin BASICin tunnistamia virhenumeroita voidaan käyttää ohjelmassa ilmoittamaan sen omista virheistä.

Aiheeseen liittyvät avainsanat: ON ERROR, ERR, ERL

EVERY

EVERY ´kokonaislukulauseke´[,´kokonaislukulauseke´] GOSUB ´rivinumero´

EVERY 500,2 GOSUB 50

KÄSKY: CPC 464:ssä on reaaliaikainen kello. BASIC-ohjelmasta voidaan kutsua aliohjelmia säännöllisin välein EVERY-käskyllä. Käytettävissä on neljä viiveajastinta, jotka määritellään EVERY-käskylausekkeen toisella ´kokonaislukulausekkeella´, joka voi olla väliltä 0 - 3. Kuhunkin viiveajastimeen voi liittyä oma aliohjelmansa.

Aiheeseen liittyvät avainsanat: AFTER, REMAIN

EXP

EXP(´numerolauseke´)

```
PRINT EXP(6.876)
 968.743625
```

FUNKTIO: Laskee "E":n arvon korotettuna kohdassa ´numerolauseke´ määriteltyyn potenssiin. E on noin 2,7182818, jonka luonnollinen logaritmi on 1.

Aiheeseen liittyvät avainsanat: LOG

FIX

FIX(´numerolauseke´)

```
PRINT FIX(9.99999)
 9
```

FUNKTIO: CINT-funktiosta poiketen FIX-funktio poistaa yksinkertaisesti ´numerolausekkeen´ desimaalipisteen oikealla puolella olevan osan jättäen jäljelle kokonaisosan ja pyöristäen sen alaspäin.

Aiheeseen liittyvät avainsanat: CINT, INT, ROUND

FOR...TO...STEP

FOR ´yksinkertainen muuttuja´ = ´alku´ TO ´loppu´ [STEP ´askelväli´]

```
FOR DAY=1 TO 5 STEP 2
```

KÄSKYJÄ: FOR...TO-silmukalla suoritetaan ohjelman runko-osa niin monta kertaa kuin halutaan. Ohjausmuuttujaa askellutetaan alku- ja loppuarvojen välillä. Askelvälin (STEP) oletusarvo on 1, ellei sitä ole erikseen määritelty.

Aiheeseen liittyvät avainsanat: NEXT, WHILE

FRE

FRE(´numerolauseke´)
tai
FRE(´merkkijonolauseke´)

```
PRINT FRE(0)
PRINT FRE("")
```

FUNKTIO: FRE-funktiolla saadaan selville, kuinka paljon ohjelmointiin käytettävää BASIC-muistia on vielä vapaana. Muoto FRE("") aiheuttaa muistinsiivouksen (garbage collection) ennen kuin ilmoittaa vapaana olevan muistitilan.

GOSUB

GOSUB 'rivinumero'

GOSUB 210

KÄSKY: GOSUB-käsky kutsuu BASIC-aliohjelmaa haarautumalla kohdassa 'rivinumero' määritellylle rivinumerolle. Katso käskyä RETURN.

Aiheeseen liittyvät avainsanat: RETURN

GOTO

GOTO 'rivinumero'

GOTO 90

KÄSKY: GOTO-käskyllä haaraudutaan määritellylle rivinumerolle. GOTO-käsky voidaan antaa myös suoraan näppäimistöltä esim. GOTO 100, jolloin ohjelman suoritus alkaa riviltä 100. Tämä eroaa RUN-käskystä sillä, ettei GOTO-käsky nollaa muuttujia. GOTO-käsky voidaan jättää tietyissä tilanteissa pois. Esim. lauseke IF A>B THEN 100 toimii samalla tavalla kuin IF A>B THEN GOTO 100.

Aiheeseen liittyvät avainsanat: GOSUB, ON GOSUB, ON GOTO, RUN, IF, THEN, ELSE

HEX$

HEX$('etumerkitön kokonaislukulauseke')[,'kokonaislukulauseke']

```
PRINT HEX$(65534)        tai        PRINT HEX$(18,4)
FFFE                                0012
```

FUNKTIO: HEX$-funktio muuttaa annetun luvun heksadesimaalimuotoon (katso liitettä II).

Aiheeseen liittyvät avainsanat: BIN$, STR$

HIMEM

HIMEM

```
?HIMEM
 43903
```

MUUTTUJA: HIMEM-muuttujalla saadaan BASIC-ohjelman käyttämän korkeimman muistitavun osoite, ja sitä voidaan käyttää numerolausekkeessa tavalliseen tapaan.

Ennen korkeimman käytettävissä olevan muistitavun nollaamista MEMORY-käskyllä kannattaa antaa käsky mm=HIMEM. Tällöin on mahdollista palata aikaisempaan muistikapasiteettiin antamalla käsky MEMORY mm.

Aiheeseen liittyvät avainsanat: FRE, MEMORY

IF...THEN...ELSE

IF ´looginen lauseke´ THEN ´valinnainen osa´ [ELSE ´valinnainen osa´]
tai
IF ´looginen lauseke´ GOTO ´rivinumero´ [ELSE ´valinnainen osa´]

IF A>B THEN A=C ELSE A=D

IF A>B GOTO 1000 ELSE 300

KÄSKYJÄ: IF-käskyä käytetään varsin laajalti BASICissa määrittelemään ohjelman haarautumiskohtia erilaisilla ehtolausekkeilla. IF-käskyn kohdatessaan tietokone laskee loogisen lausekkeen arvon, ja jos tulos on tosi, niin seuraavaksi suoritetaan THEN- tai GOTO-osa. Jos laskelman tulos on epätosi, ohjelma hyppää ELSE-osaan tai siirtyy yksinkertaisesti seuraavalle riville. Lausekkeita voidaan asettaa sisäkkäin kuinka moneen kerrokseen hyvänsä. Ainoa rajoitus on rivin pituus. IF-lauseke päätetään rivin lopulla. Samalla rivillä ei ole sallittua käyttää IF-lausekkeesta riippumattomia muita lausekkeita.

Aiheeseen liittyvät avainsanat: GOTO, OR, AND, WHILE

INK

INK ´muste´,´väri´[,´väri´]

INK 0,1

KÄSKY: Kullakin hetkellä on käytettävissä tietty määrä "musteita" senhetkisestä näyttömuodosta (MODE) riippuen (katso lukua 5). Väri tai värit voidaan vaihtaa INK-käskyllä liitteessä VI olevanväritaulukon arvojen mukaisesti.

Aiheeseen liittyvät avainsanat: PEN, PAPER

INKEY

INKEY(´kokonaislukulauseke´)

```
10 CLS:IF INKEY(55)=32 THEN 30 ELSE 20
20 CLS:GOTO 10
30 PRINT "Painat SHIFT- ja V-näppäimiä"
40 FOR t=1 TO 1000:NEXT:GOTO 10
```

FUNKTIO: INKEY-funktiolla voidaan tutkia, mitä näppäimistön näppäimiä parhaillaan painetaan. CPC 464 tutkii näppäimistönsä tilaa 0,02 sekunnin välein. INKEY-funktio on varsin käyttökelpoinen sellaisissa ohjelmissa, joissa käytetään KYLLÄ/EI-tyyppisiä vastauksia (Y/N tai K/E), koska vastausvaihtoehto voidaan valita siten, ettei [SHIFT]-näppäintä tarvitse painaa. Yllä oleva esimerkkiohjelma tulostaa ruutuun rivillä 30 olevan ilmoituksen, jos [SHIFT]- ja V-näppäimiä painetaan yhtäaikaa. (V-näppäimen numero on 55; katso liitettä III). [SHIFT]-, [CTRL]- ja muiden näppäimistön näppäinten tilan tutkiminen tapahtuu seuraavan "totuustaulukon" mukaan:

Tutkimuksen tuloksena saatava arvo	SHIFT- näppäin	CTRL- näppäin	Muu näppäin
-1	?	?	YLHÄÄLLÄ
0	YLHÄÄLLÄ	YLHÄÄLLÄ	ALHAALLA
32	ALHAALLA	YLHÄÄLLÄ	ALHAALLA
128	YLHÄÄLLÄ	ALHAALLA	ALHAALLA
160	ALHAALLA	ALHAALLA	ALHAALLA

Aiheeseen liittyvät avainsanat: INPUT, INKEY$

INKEY$

INKEY$

```
10 CLS
20 PRINT "Oletko viisas (k tai e) ?"
30 a$=INKEY$: IF a$="" GOTO 30
40 IF a$="K" OR a$="k" THEN PRINT "Varma
sti, koska olet ostanut CPC 464:n!":END
50 IF a$="E" OR A$="e" THEN PRINT"Taidat
 olla turhan vaatimaton!!!":END
60 GOTO 20
```

FUNKTIO: INKEY$-funktiolla tutkitaan, mitä näppäintä käyttäjä painaa. Tällä tavalla saadaan esim. sopivia vastauksia ohjelmaan tarvitsematta painaa [ENTER]-näppäintä vastauksen jälkeen. Funktio suorittaa halutun toiminnon, jos ohjelmassa määriteltyä näppäintä painetaan. Jos mitään näppäintä ei paineta, ohjelma kiertää tyhjää silmukkaa, kunnes se havaitsee ohjelmassa määritellyn näppäimen painalluksen ja ryhtyy toimenpiteisiin.

Aiheeseen liittyvät avainsanat: INPUT, INKEY

INP

INP('portin numero')

PRINT INP(&FF77)

FUNKTIO: INP-funktiolla saadaan osoiteosalla määritellyn I/O-portin (syöttö/tulostusportin) syöttöarvo.

Aiheeseen liittyvät avainsanat: OUT, WAIT

INPUT

INPUT[#'reittilauseke',][;]['lainausmerkeillä erotettu merkkijono';]['joukko muuttujia']
tai
INPUT[#'reittilauseke',][;]['lainausmerkeillä erotettu merkkijono',]['joukko muuttujia']

```
10 CLS
20 INPUT "Anna kaksi pilkulla erotettua
lukua ",A,B
30 IF A=B THEN PRINT "Luvut ovat yhtäsuu
ret"
40 IF A>B THEN PRINT A "on suurempi kuin
" B
50 IF A<B THEN PRINT A "on pienempi kuin
" B
60 CLEAR:GOTO 20
```

KÄSKY: INPUT-käskyllä luetaan tietoja 'reittilausekkeessa' määritellyltä reitiltä. Heti INPUT-käskyn perään laitettava puolipiste ; estää rivin vaihtumisen, kun [ENTER]-näppäintä painetaan vastaukseksi annettavien lukujen näppäilyn jälkeen. Jos kohdan 'lainausmerkeillä erotettu merkkijono' jälkeen laitetaan puolipiste ; niin INPUT-lausekkeen perään ilmestyy kysymysmerkki. Jos kysymysmerkkiä ei haluta, laitetaan puolipisteen tilalle pilkku. Jos INPUT-lausekkeen esittämään kysymykseen vastataan vääräntyyppisellä tiedolla (jos esim. numeron **0** sijaan näppäillään iso kirjain **O** , ilmestyy näyttöön viesti:

```
?Redo from start
```

Tämä tarkoittaa sitä, ettei BASIC ole ymmärtänyt vastausta ja haluaa, että vastaat kysymykseen uudelleen. Niinpä ruutuun ilmestyykin alkuperäinen kysymys uudestaan.

Kaikki INPUT-lausekkeisiin annettavat vastaukset on päätettävä [ENTER]-näppäimen painalluksella. Välittömästi kohdan 'reittilauseke' jälkeen laitettava puolipiste estää rivinvaihdon sen jälkeen kun INPUT-lausekkeen esittämään kysymykseen annettava vastaus päätetään [ENTER]-näppäimellä. Tällöin kohdistin jää juuri syötetyn tekstin perään. Jos kohdassa 'reittilauseke' määritellään kasetin syöttöreitti, ei kuvaruutuun ilmesty viestiä. Jos kasettireitti on määritelty, ei kasetin ohjausohjelmisto huomioi sitä, joten sama ohjelma voidaan lukea kummalta hyvänsä reitiltä.

Ohjelma lukee tiedostosta yhden osan kutakin kohdassa 'joukko muuttujia' määriteltyä muuttujaa kohden. Tiedon on oltava yhteensopivaa INPUT-käskyssä määritellyn tyypin kanssa, eli sen on oltava numeromuuttuja, joka päätetään pilkulla, rivinvaihdolla, tyhjällä muistialueella (white space) tai tiedoston lopulla (end of file). Viimeisen muistialueen jälkeen lähetettävät pilkut ja [ENTER]-näppäimen painallukset jäävät huomioimatta. Lainausmerkeillä erotetut merkkijonot luetaan sanatarkasti jälkimmäisiin lainausmerkkeihin asti, eikä sen jälkeen tulevia tietoja enää huomioida numeroarvojen osalta. Lainausmerkeillä erottamattomat merkkijonot päätetään samalla tavalla kuin numeroarvojen tapauksessakin.

Aiheeseen liittyvät avainsanat: LINE INPUT, READ, INKEY$

INSTR

INSTR(['kokonaislukulauseke',]'merkkijonolauseke','merkkijonolauseke')

```
PRINT INSTR(3,"BANAANI","AA")
 4
```

FUNKTIO: INSTR-funktiolla voidaan etsiä, missä kohdassa jälkimmäinen merkkijonolauseke esiintyy ensimmäisen kerran ensimmäisessä merkkijonolausekkeessa. Ennen ensimmäistä merkkijonolauseketta annettava pilkulla erotettava luku osoittaa, mistä kohdasta etsintä aloitetaan. Ellei lukua määritellä, etsintä aloitetaan ensimmäisen merkkijonolausekkeen ensimmäisestä merkistä.

Aiheeseen liittyvät avainsanat: MID$, LEFT$, RIGHT$

INT

INT('numerolauseke')

```
PRINT INT(-1.995)
-2
tai
PRINT INT(1.995)
 1
```

FUNKTIO: INT-funktio pyöristää luvun lähimpään pienempään kokonaislukuun poistaen mahdollisen desimaaliosan. INT-funktio toimii samalla tavalla kuin FIX-funktio positiivisille luvuille, mutta se antaa vastaukseksi yhtä numeroa pienemmän luvun kuin FIX-funktio negatiivisille luvuille, jos ne eivät jo ole kokonaislukuja.

Aiheeseen liittyvät avainsanat: CINT, FIX, ROUND

JOY

JOY('kokonaislukulauseke')

10 IF JOY(0)=8 THEN GOTO 100

FUNKTIO: JOY-funktiolla voidaan lukea kohdassa 'kokonaislukulauseke' (joka voi olla joko 0 tai 1) määritellyn sauvaohjaimen tila. Tila saadaan binäärimuodossa alla olevan taulukon mukaisesti:

Bitti		Desimaalimuoto
0:	Ylös	1
1:	Alas	2
2:	Vasemmalle	4
3:	Oikealle	8
4:	Tulitus 2	16
5:	Tulitus 1	32

Aiheeseen liittyvät avainsanat: INKEY

KEY

KEY 'kokonaislukulauseke',[CHR$(n)+]'merkkijonolauseke'[+CHR$(n)]

KEY 140,"RUN"+CHR$(13)

KÄSKY: KEY-käskyllä määritellään näppäimelle uusi toiminta. CPC 464:n näppäimistössä on 32 "laajennusmerkkiä" välillä 128 - 159. Ne on lueteltu liitteessä III. Kunkin tällaisen näppäimen "alle" voidaan ohjelmoida enintään 32 merkkiä käsittävä merkkijonolauseke, joka voi olla esimerkiksi joukko perättäisiä kaksoispisteellä erotettuja käskyjä. Laajennusmerkkien kokonaismäärä ei voi olla suurempi kuin 120.

Aiheeseen liittyvät avainsanat: KEY DEF

KEY DEF

KEY DEF 'näppäimen numero','toisto'[,'normaali'[,SHIFT-näppäimellä'[,'CTRL-näppäimellä']]]

KEY DEF 46,1,63

KÄSKY: Tällä käskyllä voidaan muuttaa millä hyvänsä näppäimistön näppäimellä muodostettava merkki liitteessä III määritellyllä tavalla. Yllä olevassa esimerkissä saadaan n-näppäimen (näppäin nro 46) painalluksella kysymysmerkki **?** , joka on merkki numero 63.

Näppäimen toiminta saadaan palautetuksi normaaliksi seuraavalla käskylausekkeella:

KEY DEF 46,1,110

Tässä pieni n-kirjain on merkki numero 110.

Aiheeseen liittyvät avainsanat: KEY

LEFT$

LEFT$(´merkkijonolauseke´,´kokonaislukulauseke´)

```
10 CLS
20 A$ = "AMSTRAD"
30 B$ = LEFT$(A$,3)
40 PRINT B$
RUN
(ruutu tyhjenee)
AMS
Ready
```

FUNKTIO: Tällä funktiolla erotetaan määritellystä merkkijonolausekkeesta kohdan ´kokonaislukulauseke´ määräämä määrä merkkejä kokonaislukulausekkeen määräämästä kohdasta vasemmalle (kyseinen kohta mukaan luettuna). Vastaukseksi saadaan koko ´merkkijonolauseke´, jos ´merkkijonolauseke´ on lyhyempi kuin määritelty pituus.

Aiheeseen liittyvät avainsanat: MID$, RIGHT$

LEN

LEN(´merkkijonolauseke´)

```
A$="AMSTRAD":PRINT LEN(A$)
 7
```

FUNKTIO: LEN-funktiolla saadaan selville kohdassa ´merkkijonolauseke´ olevien merkkien lukumäärä. Merkeiksi lasketaan myös välilyönnit.

LET

LET ´looginen lauseke´

```
LET x=100
```

KÄSKY: LET-käsky on jäänne BASIC-kielen alkuvaiheista, joissa muuttujien arvot piti määritellä LET-käskyllä. Käskyä saatetaan tarvita vain sovitettaessa eri tietokonemerkkien ohjelmia toisille tietokoneille. AMSTRADin BASICissa LET-käskyä ei tarvitse käyttää, joten yllä oleva esimerkki näppäillään:

```
x=100
```

LINE INPUT

LINE INPUT 'merkkijonomuuttuja'
tai
LINE INPUT [#'reittilauseke',][;]['lainausmerkeillä erotettu merkkijono';]'merkkijonomuuttuja'

LINE INPUT A$
LINE INPUT "NAME";N$

KÄSKY: Tällä käskyllä luetaan kokonainen rivi kohdassa 'reittilauseke' määritellyltä reitiltä. Heti käskyn jälkeen laitettava puolipiste ; estää rivinvaihdon/rivinsiirron kaiun. Reittilausekkeen oletusarvo on #0 eli konsoli kuten yleensäkin.

Aiheeseen liittyvät avainsanat: READ, INPUT, INKEY$

LIST

LIST ['rivinumeroalue'][,#'reittilauseke']

LIST 100-1000,#1

KÄSKY: Tällä käskyllä voidaan tulostaa ohjelmarivit määritellylle tulostusreitille. Kuvaruudun oletusarvo on #0 ja #8 on kirjoittimen tulostusreitti. Listaus voidaan keskeyttää painamalla [ESC]-näppäintä kerran, ja listaus saadaan käynnistetyksi uudelleen painamalla välilyöntinäppäintä. BASIC palaa suoraan toimintatilaan, jos [ESC]-näppäintä painetaan kahdesti peräkkäin. Ohjelmalistauksia voidaan tulostaa aiemmin määriteltyihin ikkunoihin, jolloin virheiden etsintä ohjelmista käy helpommin, koska koko kuvaruudun aluetta ei tarvitse peittää listauksella. Listaus voidaan pyytää koko ohjelmasta, ohjelman alusta tiettyyn rivinumeroon tai tietystä rivinumerosta ohjelman loppuun antamalla rivinumerot esimerkiksi seuraavasti:

LIST -100 tai LIST 30-

LOAD

LOAD 'tiedoston nimi'[,'osoitelauseke']

LOAD "LASKELMA"

KÄSKY: LOAD-käskyllä luetaan BASIC-ohjelma kasetilta muistiin, jolloin muistissa aiemmin ollut ohjelma tuhoutuu. Käskyllä voidaan ladata myös binäärimuotoinen tiedosto muistiin käyttämällä mahdollisuutta 'osoitelauseke'. Katso lukua 2.

Aiheeseen liittyvät avainsanat: RUN

LOCATE

LOCATE [#´reittilauseke´,]´x-koordinaatti´,´y-koordinaatti´

```
10 MODE 1
20 LOCATE 20,12
30 PRINT CHR$(249)
```

KÄSKY: LOCATE-käsky siirtää tekstikohdistimen x- ja y-koordinaattien määrittelemään paikkaan. Uusi paikka määritellään senhetkisen ikkunan (window) alkupisteeseen nähden. Ikkunan vasemman yläkulman koordinaatit ovat 1,1 joka on samalla oletusarvo, jos ikkuna käsittää koko kuvaruudun.

Aiheeseen liittyvät avainsanat: WINDOW

LOG

LOG(´numerolauseke´)

```
?LOG(9999)
 9.21024037
```

FUNKTIO: LOG-funktio laskee kohdassa ´numerolauseke´ määritellyn luvun luonnollisen logaritmin ja antaa reaalimuotoisen vastauksen vaikka ´numerolausekkeesta´ tulisikin kokonaisluku.

Aiheeseen liittyvät avainsanat: EXP, LOG10

LOG10

LOG10(´numerolauseke´)

```
?LOG10(9999)
 3.99995657
```

FUNKTIO: LOG10-funktio laskee kohdassa ´numerolauseke´ määritellyn luvun kymmenjärjestelmän logaritimin ja antaa vastauksen reaalimuodossa vaikka vastaukseksi tulisi kokonaisluku.

Aiheeseen liittyvät avainsanat: EXP, LOG

LOWER$

LOWER$(´merkkijonolauseke´)

```
A$="AMSTRAD":PRINT LOWER$(A$)
amstrad
```

FUNKTIO: LOWER$-funktio muuttaa kohdassa ´merkkijonolauseke´ määritellyn merkkijonon samaksi merkkijonoksi, mutta se muuttaa isot kirjaimet pieniksi kirjaimiksi niissä kohdissa, joissa aiemmin oli käytetty isoja kirjaimia. Funktio on käyttökelpoinen käsiteltäessä syöttötietoja, jossa vastaukset voivat tulla sekaisin joko suurilla tai pienillä kirjaimilla.

Aiheeseen liittyvät avainsanat: UPPER$

MAX

MAX(´joukko numerolausekkeita´)

```
10 n=66
20 PRINT MAX(1,n,3,6,4,3)
```

FUNKTIO: MAX-funktio etsii numerolausekejoukon suurimman arvon.

Aiheeseen liittyvät avainsanat: MIN

MEMORY

MEMORY ´osoitelauseke´

MEMORY &20AA

KÄSKY: Nollaa BASIC-muistin parametrit käytettävissä olevan muistikapasiteetin vaihtamiseksi asettamalla korkeimman tavun osoitteen. Katso HIMEM-avainsanan kuvausta. Muistitilan tutkimiseen käytetään käskyä FRE.

Aiheeseen liittyvät avainsanat: HIMEM, FRE

MERGE

MERGE [´tiedoston nimi´]

MERGE "SUUNNITELMA"

KÄSKY: MERGE-käsky limittää kasetilta saatavan ohjelman tietokoneen muistissa parhaillaan olevaan ohjelmaan. Se siis lisää tiedoston sisällön ohjelmamuistissa olevaan ohjelmaan. Jos MERGE-käskyyn ei määritellä tiedoston nimeä, niin BASIC yrittää limittää ohjelmamuistissa olevaan ohjelmaan ensimmäisen tiedoston, joka nauhalla tulee vastaan. Jos tiedoston nimen ensimmäinen merkki on huutomerkki **!** , niin BASIC ottaa huutomerkin pois tiedoston nimestä eikä tulosta ruutuun tavanomaisia viestejä, jotka muuten näkyisivät ruudussa luettaessa ohjelmaa kasetilta muistiin.

Jos haluat limittää kasetilla olevan ohjelman muistissa olevaan ohjelmaan etkä halua, että uusi ohjelma kirjoittuu vanhan ohjelman päälle, niin ohjelmamuistissa olevan ohjelman rivit on numeroitava uudelleen RENUM-käskyllä alkamaan niin suuresta rivinumerosta, ettei kasetilta tulevan ohjelman rivinumerot ylety niin pitkälle.

Limitettäessä hylätään kaikki muuttujat, käyttäjän toiminnot ja avoimet tiedostot. ON ERROR GOTO kytketään pois toiminnasta, RESTORE (palautus) suoritetaan ja DEFINT-, DEFREAL- & DEFSTR-asetukset palautetaan alkuarvoihinsa. Suojattuja tiedostoja ei voi limittää.

Aiheeseen liittyvät avainsanat: LOAD, CHAIN, CHAIN MERGE

MID$

MID$('merkkijono','kokonaislukulauseke'[,'kokonaislukulauseke'])

A$="AMSTRAD":PRINT MID$(A$,2,4) tai A$="AMSTRAD":b$=MID$(A$,2,4):PRINTb$
MSTR MSTR

KÄSKY ja FUNKTIO: MID$ määrittelee merkkijonosta osan (osamerkkijonon), jota voidaan käyttää joko lausekkeen kohteena (MID$ toimii käskynä) tai argumenttina merkkijonolausekkeessa (MID$ toimii funktiona). Ensimmäinen 'kokonaislukulauseke' määrittelee osamerkkijonon ensimmäisen merkin paikan. Toinen 'kokonaislukulauseke' määrittelee vastaukseksi haluttavan osamerkkijonon pituuden. Ellei sitä määritellä, saadaan vastaukseksi koko alkuperäisen merkkijonon loppuosa.

Aiheeseen liittyvät avainsanat: LEFT$, RIGHT$

MIN

MIN('joukko numerolausekkeita')

```
PRINT MIN(3,6,2.999,8,9)
 2.999
```

FUNKTIO: MIN-funktio toimii päinvastoin kuin MAX-funktio.

Aiheeseen liittyvät avainsanat: MAX

MODE

MODE 'kokonaislukulauseke'

MODE 1

KÄSKY: MODE-käskyllä valitaan haluttu näyttömuoto (0, 1 tai 2) ja tyhjennetään kuvaruutu INK 0-väriseksi, joka voi olla muukin kuin senhetkinen paperin musteen (PAPER INK) väri. Kaikki teksti- ja grafiikkaikkunat palautuvat koko kuvaruudun kokoisiksi, ja teksti- sekä grafiikkakohdistimet palautetaan omiin alkupisteisiinsä (origin).

Aiheeseen liittyvät avainsanat: WINDOW, ORIGIN

MOVE

MOVE 'x-koordinaatti','y-koordinaatti'

MOVE 34,34

KÄSKY: Siirtää grafiikkakohdistimen absoluuttisten x- ja y-koordinaattien määräämään paikkaan. Grafiikkakohdistimen x- ja y-koordinaattien arvot saadaan selville funktioilla XPOS ja YPOS.

Aiheeseen liittyvät avainsanat: MOVER, PLOT, PLOTR, DRAW, DRAWR, ORIGIN, TEST, TESTR, XPOS, YPOS

MOVER

MOVER ´x-siirtymä´,´y-siirtymä´

MOVER 34,34

KÄSKY: Siirtää grafiikkakohdistimen suhteessa senhetkisiin koordinaattiarvoihin. Grafiikkakohdistimen x- ja y-koordinaattien arvot saadaan selville funktioilla XPOS ja YPOS.

Aiheeseen liittyvät avainsanat: MOVE, PLOT, PLOTR, DRAW, DRAWR, ORIGIN, TEST, TESTR, XPOS, YPOS

NEW

NEW

NEW

KÄSKY: Poistaa ohjelmamuistissa olevan ohjelman ja nollaa muuttujien arvot. NEW-käsky ei tuhoa KEY-käskyllä määriteltyjä näppäintoimintoja eikä se tyhjennä näyttöä. NEW-käskyä on syytä käyttää varsin harkitusti, koska sen vaikutus on lähes yhtä peruuttamaton kuin tietokoneen täydellinen nollaus tai virran katkaisu.

NEXT

NEXT [´joukko muuttujia´]

FOR n=1 TO 1000:NEXT

KÄSKY: FOR-silmukan loppuun on aina laitettava NEXT-käsky. Käsky ei aina välttämättä vaadi määritettä eli käskyä voidaan käyttää yllä olevan esimerkin tapaan. Toinen tapa on lisätä NEXT-käskyn perään FOR-käskyllä määritelty muuttuja, joten yllä oleva esimerkki olisi:

NEXT n

Aiheeseen liittyvät avainsanat: FOR

ON GOSUB
ON GOTO

ON ´kokonaislukulauseke´ GOSUB ´rivinumeroluettelo´
ON ´kokonaislukulauseke´ GOTO ´rivinumeroluettelo´

10 ON DAY GOSUB 100,200,300,400,500
10 ON RATE GOTO 1000,2000,3000,4000

KÄSKYJÄ: ON GOSUB aiheuttaa haarautumisen aliohjelmaan ja ON GOTO siirtymisen tietylle riville sen mukaan, mikä tulos kohdasta 'kokonaislukulauseke' saadaan. Jos tulos on esimerkiksi 1, niin silloin valitaan luettelon ensimmäinen rivinumero. Vastaavasti jos tulos on 2, valitaan toinen rivinumero, jne. Jos yllä olevan rivin 10 DAY-muuttuja saa arvon 1, haaraudutaan rivillä 100 olevaan aliohjelmaan. Muuttujan DAY arvo 2 aiheuttaisi puolestaan haarautumisen rivillä 200 olevaan aliohjelmaan, jne.

Aiheeseen liittyvät avainsanat: GOTO, GOSUB

ON BREAK GOSUB

ON BREAK GOSUB 'rivinumero'

```
10 ON BREAK GOSUB 40
20 PRINT "Ohjelma on käynnissä"
30 GOTO 20
40 CLS
50 PRINT "Ohjelma haarautuu aliohjelmaan
 kahdella ESC-näppäimen painalluksella"
60 FOR t=1 TO 2000:NEXT
70 RETURN
```

KÄSKY: Kutsuu aliohjelmaa, kun ohjelman suoritus katkaistaan painamalla [ESC]-näppäintä kahdesti.

Aiheeseen liittyvät avainsanat: ON BREAK STOP, RETURN

ON BREAK STOP

ON BREAK STOP

```
10 ON BREAK GOSUB 40
20 PRINT "Ohjelma on käynnissä"
30 GOTO 20
40 CLS
50 PRINT "Ohjelma haarautuu aliohjelmaan
 kahdella ESC-näppäimen painalluksella"
60 FOR t=1 TO 2000:NEXT
65 ON BREAK STOP
70 RETURN
```

KÄSKY: Kun ON BREAK STOP-käsky yhdistetään ON BREAK-keskeytysaliohjelmaan, se estää silmukan jatkumisen, mutta sillä ei ole mitään muuta välitöntä vaikutusta. Yllä olevaan ohjelmaan on lisätty ON BREAK STOP-käsky riville 65, jossa ON BREAK GOSUB-silmukka toimii vain kerran.

Aiheeseen liittyvät avainsanat: ON BREAK GOSUB

ON ERROR GOTO

ON ERROR GOTO ´rivinumero´

```
10 ON ERROR GOTO 60
20 CLS
30 PRINT "Jos ohjelmassa on virhe, halua
n listauk-sen nähdäkseni mikä meni piele
en"
40 FOR t=1 TO 4000:NEXT
50 GOTO 200
60 CLS:PRINT"VIRHE ON RIVILLÄ";ERL:PRINT
"
70 LIST
```

KÄSKY: Aiheuttaa siirtymisen ´rivinumerolla´ määritellylle riville, jos ohjelmassa havaitaan virhe. Edellä olevassa esimerkkiohjelmassa virhe löytyy riviltä 50.

Aiheeseen liittyvät avainsanat: ERR, ERL, RESUME

ON SQ GOSUB

ON SQ (´kanava´) GOSUB ´rivinumero´

ON SQ (2) GOSUB 2000

KÄSKY: Sallii keskeytyksen, kun määritellyn kanavan äänijonossa on vapaa rako. Kanava ilmaistaan kokonaisluvulla seuraavasti:

1: kanava A
2: kanava B
4: kanava C

Aiheeseen liittyvät avainsanat: SOUND, SQ

OPENIN

OPENIN ´tiedoston nimi´

100 OPENIN "!INFORMAATIOTA"

KÄSKY: Avaa kasetilta syöttötiedoston, joka sisältää informaatiota käytettäväksi tietokoneen muistissa parhaillaan olevassa ohjelmassa.

Jos tiedoston nimen ensimmäinen merkki on huutomerkki **!** niin ruutuun ei tulostu normaaleja kasetinkäsittelyviestejä, ja ohjelma lukee ensimmäisen lohkon (block) kasetilta valmiiksi käsittelyä varten.

Aiheeseen liittyvät avainsanat: CLOSEIN, OPENOUT

OPENOUT

OPENOUT ´tiedoston nimi´

OPENOUT "!TOSIASIAT"

KÄSKY: Avaa tulostustiedoston kasetille käytettäväksi tietokoneen muistissa parhaillaan olevassa ohjelmassa. Jos tiedoston nimen ensimmäinen merkki on ! niin normaaleja kasetinkäsittelyviestejä ei ilmesty ruutuun, ja ohjelma luo ensimmäisen tietolohkon sekä määritellyn nimen mukaisen tiedoston. Kukin lohko sisältää 2K tietoa, eikä nauhalle tallennu mitään ennen kuin 2K:n puskuri on täynnä tai tiedostot suljetaan käyttäen CLOSEOUT-käskyä tms.

HUOM! NEW-käsky hylkää mahdolliset puskurissa olevat avoimet tiedostot ja tuhoaa tiedot.

Aiheeseen liittyvät avainsanat: CLOSEOUT, OPENIN

ORIGIN

ORIGIN ´x´,´y´[,´vasen´,´oikea´,´ylin´.´alin´]

```
10 CLS:BORDER 13
20 ORIGIN 0,0,50,590,350,50
30 DRAW 540,350
40 GOTO 20
```

KÄSKY: ORIGIN-käskyllä määritellään grafiikkakohdistimen alkupiste. Hakasulkeissa oleva valinnainen osa sisältää käskyt uuden grafiikkaikkunan asettamiseksi. Ikkuna toimii kaikissa kolmessa näyttömuodossa kuva-alkion osoitustekniikasta johtuen.

Alkupiste on kohta, jossa sekä x- että y-koordinaatin arvo on 0. Tämä piste sijaitsee ikkunan vasemmassa alakulmassa. Koordinaattien arvot kasvavat oikealle (x) ja ylöspäin (y). Grafiikkaruudukoita on kuvattu tarkemmin liitteessä VI.

Jos jokin ikkunan kulmista määritellään kuvaruudun ulkopuolella olevaan kohtaan, niin tietokone olettaa kyseisen arvon tarkoittavan tässä suunnassa kauimpana "näkyvää" kohtaa.

Aiheeseen liittyvät avainsanat: WINDOW

OUT

OUT ´portin numero´,kokonaislukulauseke´

OUT &F8F4,10

KÄSKY: Lähettää kohdassa ´kokonaislukulauseke´ määritellyn arvon (väliltä 0...255) porttiin, joka on määritelty kohdassa ´portin numero´ sen osoitteella.

Aiheeseen liittyvät avainsanat: INP, WAIT

PAPER

PAPER [#'reittilauseke',]'peittomuste'

```
10 MODE 0
20 FOR p=0 TO 15
30 PAPER p:CLS
40 PEN 15-p
50 LOCATE 6,12:PRINT "PAPERI"p
60 FOR t=1 TO 500: NEXT t
70 NEXT p
```

KÄSKY: PAPER-käskyllä asetetaan taustan musteen väri merkkejä varten. Kun merkkejä kirjoitetaan tekstiruutuun, merkkisolu täytetään paperin musteen värillä ennen merkin kirjoittamista, ellei näyttötavaksi ole valittu läpinäkyvää näyttötapaa (katso sivua 110).

Näytössä käytettävien värien määrä riippuu valitusta näyttömuodosta. Jos määriteltyä 'peittomustetta' ei ole käytettävissä, tietokone valitsee paperille oletusarvon käytettävissä olevista väreistä.

Aiheeseen liittyvät avainsanat: INK, WINDOW, PEN

PEEK

PEEK('osoitelauseke')

```
10 MODE 2
20 INK 1,0: INK 0,12 : BORDER 12
30 INPUT "Tutkimuksen alkuosoite";alku
40 INPUT "Tutkimuksen loppuosoite";loppu
50 FOR n= alku TO loppu
60 arvo$=HEX$(PEEK(n),2)
70 PRINT "Osoitteessa    ";HEX$((n),4),;
"eli";n" on ";arvo$
80 NEXT
```

FUNKTIO: PEEK-funktiolla voidaan tutkia muistipaikan sisältöä. Voit tutkia CPC 464:n RAM-muistin sisällön vaikka kokonaan yllä olevalla "apuohjelmalla". Ohjelma lukee alemman ROM-lohkon (&0000-&3FFFH) ja ylemmän ROM-lohkon (&C000-&FFFF) alla olevan RAM-muistin sisältöä eikä siis ROM-muistin sisältöä.

Aiheeseen liittyvät avainsanat: POKE

PEN

PEN [#'reittilauseke',]'peittomuste'

```
PEN 1,2
```

KÄSKY: PEN-käsky määrää musteen, jota käytetään piirrettäessä määriteltyyn ruudun tulostusreittiin (ikkunaan). Oletusarvona käytetään ruudun ikkunaa #0.

Aiheeseen liittyvät avainsanat: INK, PAPER

PI

PI

```
PRINT PI
3.14159265

10 REM Perspektiivistä piirtämistä
20 MODE 2
30 RAD
40 INK 1,0
50 INK 0,12
60 BORDER 9
70 FOR N= 1 TO 200
80 ORIGIN 420,0
90 DRAW 0,200
100 REM kulmien piirtoa pakopisteestä
110 DRAW 30*N*SIN(N*PI/4),(SIN(PI/2))*N*
SIN(N)
120 NEXT
130 MOVE 0,200
140 DRAWR 0,50
150 DRAWR 40,0
160 WINDOW 1,40,1,10
170 PRINT "Nyt voit tehdä HANGMAN-ohjelm
an":PRINT "(hirsipuun) valmiiksi!"
```

FUNKTIO: PI-funktion arvo on sama kuin ympyrän kehän ja halkaisijan suhde. Tarkin koneella laskettava likiarvo on 3,141592653468251. PI-funktiota käytetään varsin laajalti yllä olevan esimerkkiohjelman kaltaisissa grafiikka-ohjelmissa.

Aiheeseen liittyvät avainsanat: DEG, RAD

PLOT

PLOT ´x-koordinaatti´,´y-koordinaatti´[,´peittomuste´]

```
10 MODE 2:PRINT "Anna 4 pilkulla erotett
ua lukua":PRINT
20 PRINT "Anna x-alkupiste (0-639), y-al
kupiste (0-399), säde ja kulmaväli":INPU
T x,y,r,s
30 ORIGIN x,y
40 FOR angle = 1 TO 360 STEP s
50 XPOINT = r*COS(angle)
60 YPOINT = r*SIN(angle)
70 PLOT XPOINT,YPOINT
74 REM MOVE 0,0
75 REM DRAW XPOINT,YPOINT
80 NEXT
```

KÄSKY: Anna luvut 320, 200, 20, 1 ensimmäiseksi vastaukseksi. PLOT-käsky on melkein sama kuin DRAW-käskykin, mutta PLOT-käsky piirtää vain koordinaatti-arvoilla määritellyn kuva-alkion. Jos poistat REM-käskyn rivin 75 alusta ja laitat REM-käskyn rivin 70 alkuun (jolloin rivi 70 ei toimi ohjelmassa), näet käskyjen vaikutuksen eron. (Poista REM-käsky rivin 74 alusta, jolloin saat ympyrän täyteen).

Ympyrän ääriviivojen täyttö tapahtuu piirtämällä ympyrän kehä useasta eri osasta yhä uudelleen ja uudelleen moneen kertaan ympyrän ympäri. Ohjelmaa ei ole palautettu radiaanien (RAD) käyttötilaan kulmien laskennassa, joten kullakin askeleella piirttyvä "kulma" on huomattavasti enemmän kuin yksi aste. Lisää ohjelmaan käskyrivi **25 DEG** ja aja ohjelma uudelleen.

Aiheeseen liittyvät avainsanat: DRAW, DRAWR, PLOTR, MOVE, MOVER, ORIGIN, TEST, TESTR, XPOS, YPOS

PLOTR

PLOTR ´x-koordinaatti´,´y-koordinaatti´[,´peittomuste´]

```
10 MODE 2:PRINT "Anna 4 pilkulla erotett
ua lukua":PRINT
20 PRINT "Anna x-alkupiste (0-639), y-al
kupiste (0-399), säde ja kulmaväli":INPU
T x,y,r,s
30 ORIGIN x,y
40 FOR angle = 1 TO 360 STEP s
50 XPOINT = r*COS(angle)
60 YPOINT = r*SIN(angle)
70 PLOTR XPOINT,YPOINT
80 NEXT:GOTO 40
```

KÄSKY: Anna numerot 320,200,20,1 ensimmäiseksi vastaukseksi. PLOTR-käsky on melkein samanlainen kuin DRAWR-käskykin, mutta PLOTR-käsky piirtää vain koordinaattiarvoilla määritellyn kuva-alkion.

Aiheeseen liittyvät avainsanat: DRAW, DRAWR, PLOT, MOVE, MOVER, ORIGIN, TEST, TESTR, XPOS, YPOS

POKE

POKE ´osoitelauseke´,´kokonaislukulauseke´

POKE &00FF,10

KÄSKY: POKE-käsky mahdollistaa suoran pääsyn tietokoneen muistiin ja sillä voidaan ladata väliltä 0...255 oleva ´kokonaislukulauseke´ suoraan kohdassa ´osoitelauseke´ määriteltyyn osoitteeseen. POKE-käskyn kanssa on syytä olla varovainen, koska järjestelmä menee helposti sekaisin jos POKE-käskyä käytetään miten sattuu.

Aiheeseen liittyvät avainsanat: PEEK

POS

POS(#´reittilauseke´)

```
PRINT POS(#0)
 1
```

FUNKTIO: POS-funktiolla saadaan selville kohdassa ´reittilauseke´ määritellyn reitin senhetkinen paikka. Tässä yhteydessä ´reittilausekkeelle´ ei ole mitään oletusarvoa ja sen pois jättäminen aiheuttaa virheilmoituksen **Syntax Error** .

Kuvaruudusta POS-funktiolla saadaan selville tekstikohdistimen senhetkinen x-koordinaatti suhteessa senhetkisen ikkunan alkupisteeseen, jossa alkupiste on vasen ylänurkka, jonka koordinaattiarvot ovat 1,1.

Kirjoittimelta POS-funktiolla saadaan selville kirjoituspään (vaunun) paikka, jossa 1 on vasen reuna. Funktion piiriin sisältyvät kaikki merkit, joiden ASCII-viitenumero on suurempi kuin &1F(31).

Kasetin tulostusreitillä POS-funktio toimii kuten kirjoittimellakin.

Aiheeseen liittyvät avainsanat: VPOS

PRINT

PRINT [#´reittilauseke´,][´tulostettava joukko´][´USING-lauseke´][´erotin´]

```
PRINT #0,"abc"
```

´tulostettava joukko´ on: ´lauseke´[´erotin´´tulostettava alkio´]*
´tulostettava alkio´ on: ´lauseke´

tai

PRINT SPC(´kokonaislukulauseke´)
PRINT TAB(´kokonaislukulauseke´)

Tietoja kirjoitetaan kasettitiedostoon seuraavasti:

```
10 OPENOUT "DATA"
20 PRINT #9, "Hellurei"
30 CLOSEOUT
```

Tietoja kirjoitetaan rivikirjoittimelle seuraavasti:

```
10 PALKKA = 2000*PI
20 PRINT #8, USING "####.##";PALKKA
30 PRINT #0, PALKKA
RUN
 6283.18531
Ready
```

Samalla kirjoitin tulostaa:

```
 6283.19
```

KÄSKY: PRINT-käskyllä tulostetaan tietoja 'reittilausekkeessa' määritellylle reitille käyttäen 'USING-lausekkeessa' määriteltyä muotoa. Muotoiluun käytettävät merkit löytyvät seuraavalla sivulla olevasta taulukosta. AMSTRADin BASICissa on mahdollista käyttää kaikkia vakiintuneita PRINT-lausekkeen muotoilutapoja.

Jos muotoilua ei erikseen määritellä, BASIC tulostaa vapaasti muotoillen, jolloin 'tulostettavan alkion' perässä oleva pilkku siirtää tulostettavan alkion seuraavan tulostusvyöhykkeen (ZONE) alkuun (vyöhykkeen leveyden oletusarvo on 13 merkkiä). Puolipisteellä erotetaan lausekkeet, joissa käytetään yhtä välilyöntiä.

Käsky PRINT SPC ('kokonaislukulauseke') tulostaa halutun määrän välilyöntejä. Oletusarvo on 0, jos 'numerolauseke' on negatiivinen. Jos lauseke on suurempi kuin tulostuslaitteen (kuvaruudun, kirjoittimen, tms.) leveys, niin se pienennetään tulostusleveyteen sopivaksi. SPC-käskyä ei tarvitse päättää pilkulla tai puolipisteellä, sillä oletusarvona käytetään puolipistettä kaikissa yhteyksissä.

Käsky PRINT TAB ('kokonaislukulauseke') tulostaa välilyöntejä, kunnes ollaan 'kokonaislukulausekkeessa' määritellyssä tulostuskohdassa. Oletusarvona käytetään arvoa 1, jos annettava arvo on negatiivinen. Myös TAB-käskyllä annettavien välilyöntien määrää supistetaan SPC-käskyn tapaan, ellei tulostus mahdu käytettävissä olevaan tulostusleveyteen.

Jos haluttu kohta on sama tai suurempi kuin senhetkinen kohta niin BASIC tulostaa välilyöntejä, kunnes ollaan halutussa kohdassa. Jos haluttu kohta on taas pienempi kuin senhetkinen kohta, aiheuttaa se vaunun palautuksen. Tämän jälkeen tulostetaan välilyöntejä, kunnes ollaan määritellyssä kohdassa. Tietokone olettaa, että TAB-käsky päättyy puolipisteellä kaikissa yhteyksissä.

Aiheeseen liittyvät avainsanat: USING, TAB, SPC

PRINT USING "´muotoilukenttien määrittelymerkit´"

Numerot

Määrittely-merkki	**Mahdolliset numerot**	**Kenttä-merkit**	**Määritelmä**	**Esimerkki**
#	1	1	Numerokenttä	####
.	0	1	Desimaalipiste	#.#
+	0	1	Etumerkki luvun eteen tai perään	+##
			Positiivisen luvun perään +	###+
			Negatiivisten lukujen edessä ei voi olla merkkiä -	
-	0	1	Etumerkki luvun perään. Tulostaa merkin - jos luku on negatiivinen, muuten tyhjän	##.##-
**	2	2	Tähtimerkit eteen	**###.##
$$	1	2	Ensimmäisen numeron eteen laitetaan liukuva dollarinmerkki	$$##.##
**$	2	3	Täyttö tähtimerkeillä ja liukuvalla dollarinmerkillä	**$#.##
,	1	1	Tulostaa pilkun joka kolmannen numeron jälkeen (vain desimaalipisteestä vasemmalle)	##,###,##
↑↑↑↑	0	4	Eksponenttimuoto. Luvun tulostus kohdistetaan nollasta poikkeavan etunumeron mukaan	#,##↑↑↑

Merkkijonot

!	Vain ensimmäinen merkki	!
´välilyöntejä´	Määriteltyjen välilyöntien lukumäärä sekä yksi välilyönti merkkikentän alkuun ja loppuun	
&	Muuttuva pituuskenttä	&

RAD

RAD

RAD

KÄSKY: RAD-käsky asettaa radiaanien käyttötilan.

Aiheeseen liittyvät avainsanat: DEG, SIN, COS, TAN, ATN

RANDOMIZE

RANDOMIZE ['numerolauseke']

```
10 RANDOMIZE 23
20 PRINT RND(6)
```

KÄSKY: BASICin satunnaislukugeneraattori tuottaa näennäisesti joukon perättäisiä satunnaislukuja, jossa kukin luku riippuu edellisestä luvusta. Lukujen muodostaminen aloitetaan kohdassa 'numerolauseke' määritellystä luvusta, ja järjestys on aina sama. RANDOMIZE-käskyllä asetetaan satunnaislukugeneraattorille uusi alkuarvo, joka voi olla kohdassa 'numerolauseke' määritelty arvo tai se annetaan näppäimistöltä. Käskyllä RANDOMIZE TIME saadaan muodostetuksi jo varsin sattumanvarainen lukujoukko.

Aiheeseen liittyvät avainsanat: RND

READ

READ 'joukko muuttujia'

```
10 FOR X=1 TO 4
20 READ N$
30 PRINT N$
40 DATA TUOMAS,JANNE,EIJA,JUKKA
50 NEXT
```

KÄSKY: READ-käsky lukee tietoja joukosta vakioita, jotka on sijoitettu vastaavaan DATA-lausekkeeseen. Lisäksi READ-käsky liittää tiedot vastaaviin muuttujiin ja siirtyy automaattisesti seuraavan DATA-lausekkeessa olevan alkion kohdalle. RESTORE-käsky palauttaa osoittimen DATA-lausekkeen alkuun. Katso avainsanaa DATA.

Aiheeseen liittyvät avainsanat: DATA, RESTORE

RELEASE

RELEASE ´äänikanava´

RELEASE 4

KÄSKY: Kun ääni laitetaan äänikanavan jonoon, siihen voi sisältyä "pitotila" (hold). Kanavat vapautetaan RELEASE-käskyllä, jos jokin määritellyistä kanavista on pitotilassa. Äänikanavan tunnistuslauseke on binäärimuotoinen: A-kanava = bitti 0, B-kanava = bitti 1 ja C-kanava = bitti 2. Täten 4 (binäärimuodossa 0100) vapauttaa jonon C.

Aiheeseen liittyvät avainsanat: SOUND

REM

REM ´rivin loppuosa´

10 REM Huomautuksia tai selityksiä

KÄSKY: REM-käskyä käytetään silloin, kun ohjelmaan halutaan laittaa huomautuksia tai selityksiä. REM-käskyn jälkeen samalla rivillä olevat käskyt samoin kuin huomautukset ja selitykset eivät vaikuta ohjelman toimintaan millään tavalla. Ohjelma ei myöskään huomioi rivillä mahdollisesti olevaa kaksoispistettä vaan ohjelma jättää kaiken huomioimatta REM-käskystä rivin loppuun. Rivillä oleva yksittäinen lainausmerkki ' (joka ei kuulu osana ´merkkisanalausekkeeseen´) vastaa muotoa :REM. Tämä ei päde DATA-lausekkeen yhteydessä, jossa lainausmerkkiä käsitellään tietyissä tapauksissa merkkijonon osana.

REMAIN

REMAIN (´kokonaislukulauseke´)

REMAIN (3)
PRINT #6,REMAIN(0);

FUNKTIO: Estää määritellyn viiveajastimen toiminnan (ajastimia on 4 välillä 0 - 3). Samalla REMAIN-käsky lukee viiveajastimen jäljellä olevan "saldon". Vastaukseksi saadaan 0, jos viiveajastimen toiminta ei ollut sallittu.

Aiheeseen liittyvät avainsanat: AFTER, EVERY.

RENUM

RENUM ['uusi rivinumero'][,['vanha rivinumero'][,'inkrementti']]

RENUM
RENUM 100,,100

KÄSKY: RENUM-käskyä käytetään ohjelmarivien uudelleennumerointiin. Uudelleennumerointi suoritetaan määritellyllä rivinumeroalueella käyttäen määriteltyä inkrementtiä eli askelväliä. 'Uusi rivinumero' määrittelee ensimmäisen uuden rivinumeron, jonka oletusarvo on 10. 'Vanha rivinumero' määrittelee kohdan, josta uudelleennumerointi on aloitettava. Ellei sitä määritellä, uudelleennumerointi alkaa ohjelman ensimmäisestä rivistä. 'Inkrementti' määrittelee rivinumeroiden välisen askelvälin, jossa oletusarvo on myös 10.

RENUM-käsky huolehtii myös kaikista GOSUB- ja GOTO-käskyistä sekä muista rivikutsuista. Annettaessa pelkkä RENUM-käsky ohjelman uudelleennumerointi tapahtuu samalla tavalla kuin jos käsky annettaisiin muodossa RENUM 10,,10.

Rivinumeroiden on oltava väliltä 1 - 65535.

RESTORE

RESTORE ['rivinumero']

RESTORE 300

```
10 FOR N=1 TO 4
20 READ A$
30 PRINT A$ " ";
40 DATA restore,palauttaa,tiedot,käyttöön
50 NEXT
60 PRINT
70 RESTORE
80 GOTO 10
```

KÄSKY: RESTORE-käsky palauttaa tietojen lukuun käytettävän osoittimen kohdassa 'rivinumero' määritellyllä rivillä olevan DATA-lausekkeen alkuun. Jos 'rivinumero' jätetään pois, osoitin palautuu takaisin ensimmäisen DATA-lausekkeen alkuun.

Aiheeseen liittyvät avainsanat: READ, DATA

RESUME

RESUME ['rivinumero']
tai
RESUME NEXT

RESUME 300

KÄSKY: Kun ohjelmassa on havaittu virhe ON ERROR GOTO-käskyllä ja virhe on käsitelty, RESUME sallii jatkaa ohjelman normaalia suorittamista. Haluttaessa voidaan myös määritellä se rivinumero, mistä ohjelman suoritus jatkuu.

Aiheeseen liittyvät avainsanat: ON ERROR GOTO

RETURN

RETURN

RETURN

KÄSKY: RETURN-käsky ilmoittaa, että aliohjelma on päättynyt. Tällöin BASIC palaa suorittamaan pääohjelmaa. Pääohjelman suoritus jatkuu siltä riviltä, jota ennen kyseisen aliohjelman GOSUB-käsky on.

Aiheeseen liittyvät avainsanat: GOSUB, ON x GOSUB, ON SQ GOSUB, AFTER n GOSUB, EVERY n GOSUB, ON BREAK GOSUB

RIGHT$

RIGHT$(´merkkijonolauseke´,´kokonaislukulauseke´)

```
10 CLS
20 A$ = "AMSTRAD"
30 B$ = RIGHT$(A$,3)
40 PRINT B$
RUN
(Ruutu tyhjenee, jonka jälkeen ruutuun ilmestyy:)
RAD
Ready
```

FUNKTIO: RIGHT$-funktio erottaa kohdassa ´merkkijonolauseke´ määritellystä merkkijonosta kohdan ´kokonaislukulauseke´ määritellyn määrän merkkejä kokonaislukulausekkeen ilmoittamasta kohdasta oikealle (kyseinen kohta mukaan luettuna). Jos ´merkkijonolauseke´ on lyhyempi kuin kokonaislukulausekkeella määritelty pituus, saadaan vastaukseksi koko ´merkkijonolauseke´.

Aiheeseen liittyvät avainsanat: MID$, LEFT$

RND

RND[(´numerolauseke´)]

```
10 RANDOMIZE 23
20 PRINT RND(6)
```

FUNKTIO: RND-funktio hakee satunnaisluvun, joka voi olla tietyn jonon seuraava luku, sama kuin edellinen luku tai uuden jonon ensimmäinen luku. Yllä olevassa ohjelmassa käytetty RANDOMIZE-käsky varmistaa sen, että RND(6)-funktio antaa vastaukseksi joka kerta saman luvun (0.146940658).

Funktio RND(0) antaa vastaukseksi kopion viimeksi muodostetusta satunnaisluvusta. Jos ´numerolauseke´ on negatiivinen, on mahdollista aavistaa vastaukseksi saatavat satunnaisluvut.

Aiheeseen liittyvät avainsanat: RANDOMIZE

ROUND

ROUND('numerolauseke'[,'kokonaislukulauseke'])

```
10 x=0.123456789
20 FOR r=9 TO 0 STEP -1:PRINT r,ROUND(x,
r):NEXT
25 x=123456789
30 FOR r=0 TO -9 STEP -1
40 PRINT r,ROUND(x,r)
50 NEXT
```

FUNKTIO: ROUND-funktio pyöristää 'numerolausekkeen' käyttäen kohdassa 'kokonaislukulauseke' olevan luvun verran desimaalinumeroita. 'Numerolauseke' voidaan pyöristää myös 'kokonaislukulausekkeen' määräämään kymmenen potenssiin. Jos 'kokonaislukulauseke' on pienempi kuin 0, niin arvo pyöristetään siten, että vastaukseksi saadaan absoluuttinen kokonaisluku, jonka perässä on 'kokonaislukulausekkeen' määrittelemä määrä nollia ennen desimaalipistettä.

Aiheeseen liittyvät avainsanat: INT, FIX, CINT, ABS

RUN

RUN 'merkkijonolauseke'

RUN "WELCOME"

KÄSKY: RUN-käsky lataa ohjelman kasetilta muistiin käynnistäen sen suorituksen. Jos 'merkkijonolauseke' on tyhjä "" niin BASIC yrittää ladata ja suorittaa ensimmäisen nauhalta löytämänsä tiedoston. Jos merkkijonolausekkeen ensimmäinen merkki on ! niin normaaleja kasetinkäsittelyilmoituksia ei ilmesty kuvaruutuun.

> HUOM! BASIC suorittaa aina NEW-käskyn avain automaattisesti lukiessaan 'tiedoston nimen' nauhalta.

Aiheeseen liittyvät avainsanat: LOAD

RUN

RUN ['rivinumero']

RUN 100

KÄSKY: Käynnistää ohjelmamuistissa olevan ohjelman suorituksen kohdassa 'rivinumero' määritellyltä riviltä tai ohjelman alusta, ellei rivinumeroa määritellä. RUN-käsky nollaa muistista kaikki ohjelmafunktioiden ja -muuttujien sekä käyttäjän funktioiden ja muuttujien arvot. DEFINT-, DEFREAL- ja DEFSTR-asetukset palautuvat alkuarvoihinsa. Kaikki kasettitiedostot hylätään ja mahdolliset puskurissa olevat tulosteet menetetään.

Aiheeseen liittyvät avainsanat: LOAD

SAVE

SAVE ´tiedoston nimi´[,´tiedoston tyyppi´][,´binääriparametrit´]

```
SAVE "OHJELMA",P
```

KÄSKY: SAVE-käsky tallentaa muistissa olevan ohjelman nauhalle nimellä ´tiedoston nimi´.
,A tallentaa ohjelman ASCII-muodossa.
,P suojaa tiedoston.
,B tallentaa muistialueen binääritiedostona, esimerkkinä vedos kuvaruudusta.

Aiheeseen liittyvät avainsanat: LOAD, RUN ´tiedoston nimi´, MERGE, CHAIN, CHAIN MERGE

SGN

SGN(´numerolauseke´)

```
10 INPUT "Mikä on pankkitilisi saldo";tu
ohet
20 IF SGN(tuohet) <1 GOTO 30 ELSE 40
30 PRINT "Voi voi!":END
40 PRINT "Sinulla on enemmän rahaa kuin
minulla."
```

FUNKTIO: SGN-funktio määrittelee ´numerolausekkeen´ etumerkin. Vastaukseksi saadaan -1, jos ´numerolauseke´ on pienempi kuin 0. Vastaukseksi saadaan 0, jos ´numerolauseke´ = 0. Vastaukseksi saadaan 1, jos ´numerolauseke´ on suurempi kuin 0.

Aiheeseen liittyvät avainsanat: ABS

SIN

SIN(´numerolauseke´)

```
PRINT SIN(PI/2)
 1
```

FUNKTIO: SIN-funktio laskee ´numerolausekkeen´ sinin todellisen arvon. Oletusarvona käytetään mittayksikkönä radiaaneja ellei muuta ole määritelty DEG-käskyllä.

Aiheeseen liittyvät avainsanat: COS, TAN, ATN, DEG, RAD

SOUND

SOUND ´kanavan tila´,´säveljakso´[,´kesto´[,´voimakkuus´[,´voimakkuusvaippa´
[,´sävelvaippa´[,´kohinajakso´]]]]]

SOUND 1,200,1000,7,0,0,1

KÄSKY: CPC 464:llä on mahdollista muodostaa varsin monipuolisia äänitehosteita BASIC-kielellä. Aihetta käsitellään laajemmin luvussa 6 sekä ENV- ja ENT-avainsanojen kohdalla.

Aiheeseen liittyvät avainsanat: ENV, ENT

SPACE$

SPACE$(´kokonaislukulauseke´)

SPACE$(190)

FUNKTIO: SPACE$-funktio muodostaa kohdassa ´kokonaislukulauseke´ määritellyn pituisen merkkijonon, joka koostuu välilyönneistä (maks. 255).

Aiheeseen liittyvät avainsanat: PRINT, SPC, TAB

SPEED INK

SPEED INK ´kokonaislukulauseke´,´kokonaislukulauseke´

```
5 INK 0,9,12:INK 1,0,26
10 BORDER 12,9
20 SPEED INK 50,20
```

KÄSKY: INK- ja BORDER-käskyt suovat mahdollisuuden liittää kaksi väriä kuhunkin musteeseen (INK), jolloin muste vuorottelee kahden värin välillä. Ensimmäinen ´kokonaislukulauseke´ määrittelee ensimmäiselle määritellylle musteelle käytettävän ajan ja toinen ´kokonaislukulauseke´ määrittelee toisen musteen ajan. Ajan mittausyksikkönä värinvaihtojen välillä käytetään 0,02 sekuntia. Värien valinnassa ja toistonopeuksissa kannattaa käyttää kokemuksen perusteella hankittuja arvoja turhan räikeiden tulosten välttämiseksi.

Aiheeseen liittyvät avainsanat: INK, BORDER

SPEED KEY

SPEED KEY ´aloitusviive´,´toistonopeus´

SPEED KEY 20,3

KÄSKY: CPC 464:n näppäimistön näppäimet toistavat automaattisesti näppäimen merkkiä, jos näppäin pidetään koko ajan alhaalla. Toisto käynnistyy kohdassa ´aloitusviive´ määritellyn ajan kuluttua ja toistonopeus määritellään kohdalla ´toistonopeus´. Arvojen määrittelyssä käytetään 0,02 sekunnin mittaisia yksikköjä, ja arvot on valittava väliltä 1...255. Oletusarvot ovat 30 ja 2 (joissakin malleissa voi olla myös 10 ja 10).

Yksittäisen merkin näppäily tuottaa hankaluuksia, jos ´aloitusviive´ valitaan liian pieneksi. Käsky ei kuitenkaan vaikuta siihen nopeuteen, jolla tietokoneen ohjelmisto "lukee" näppäimistöä.

Kaikissa näppäimistön näppäimissä ei ole toistotoimintoa. KEY DEF-käskyllä voidaan määritellä uudelleen halutut parametrit tietyille näppäimille.

Aiheeseen liittyvät avainsanat: KEY DEF

SPEED WRITE

SPEED WRITE ´kokonaislukulauseke´

SPEED WRITE 1

KÄSKY: Tietoja voidaan tallentaa kasetille joko 2000 baudin nopeudella (jolloin ´kokonaislukulauseke´ = 1) tai oletusarvona käytettävällä 1000 baudin nopeudella (jolloin ´kokonaislukulauseke´ = 0). CPC 464 valitsee automaattisesti oikean lukunopeuden tiedosto-ohjelmistosta saamiensa tietojen avulla ladattaessa ohjelma kasetilta muistiin. Käyttäjän ei siis tarvitse määritellä lukunopeutta ladattaessa ohjelma kasetilta muitiin.

Tallennuksen luotettavuus on paras mahdollinen käytettäessä 1000 baudin siirtonopeutta varsinkin sellaisilla kaseteilla, joiden nauhalaadusta ei ole varmuutta. Asiaa on käsitelty tarkemmin luvussa 2.

Aiheeseen liittyvät avainsanat: SAVE

SQ

SQ(´kanava´]

```
10 MODE 1
20 FOR n=20 TO 0 STEP -1
30 PRINT n;
40 SOUND 1,10+n,100,7
50 WHILE SQ(1)>127:WEND
60 NEXT
```

FUNKTIO: SQ-funktiota käytetään tarkistamaan määritellyn kanavan äänijonossa olevien vapaiden rakojen määrä. Kanavia A, B ja C vastaa numerot 1, 2 ja 4. Funktio määrittelee, onko kanava aktiivinen, ja ellei se ole, miksi syöttötieto jonon päässä (jos niitä on lainkaan) odottaa. Tulos saadaan binäärimuodossa:

0,1,2 osoittavat jonossa olevien vapaiden syöttökohtien lukumäärän.
3,4,5 osoittavat jonon pään (jos niitä on lainkaan) tahdistustilan (rendezvous).
6 asetetaan jos jonon päässä on pitotoiminto (HOLD).
7 asetetaan jos kanava on parhaillaan aktiivinen (toiminnassa).

Aiheeseen liittyvät avainsanat: SOUND, ON SQ GOSUB

SQR

SQR('numerolauseke')

```
PRINT SQR(9)
 3
```

FUNKTIO: Antaa 'numerolausekkeen' neliöjuuren.

Aiheeseen liittyvät avainsanat: PRINT

STOP

STOP

```
300 IF n<0 THEN STOP
```

KÄSKY: STOP-käskyä käytetään ohjelman suorituksen pysäyttämiseksi, mutta BASIC jää tilaan, josta ohjelma voidaan käynnistää uudelleen STOP-käskyn jälkeen. STOP-käskyä voidaan käyttää esim. ohjelman keskeyttämiseksi tietyssä kohdassa etsittäessä ohjelmista virheitä.

Aiheeseen liittyvät avainsanat: CONT, END

STR$

STR$('numerolauseke')

```
PRINT STR$(&766)
 1894

PRINT STR$(&X1010100)
 84
```

FUNKTIO: STR$-funktio muuntaa 'numerolausekkeen' desimaalimuotoiseksi merkkijonoksi, joka on samaa muotoa kuin PRINT-käskyssä käytettävä muotokin.

Aiheeseen liittyvät avainsanat: VAL, PRINT, HEX$, BIN$

STRING$

STRING$('kokonaislukulauseke','merkkimäärite')

```
PRINT STRING$(&16,"*")
**********************
```

FUNKTIO: STRING$-funktiolla saadaan muodostetuksi haluttu määrä merkkejä. Määrä määritellään sulkulausekkeen 'kokonaisluvulla' (joka on heksadesimaalimuodossa) ja lainausmerkeissä oleva merkki määrittelee tulostettavan merkin.

Aiheeseen liittyvät avainsanat: SPACE$

SYMBOL

SYMBOL 'merkin numero','merkinmäärittelyluvut riveittäin'

```
5 MODE 2
10 SYMBOL AFTER 90
20 SYMBOL 93,&80,&40,&20,&10,&8,&4,&2,&1
30 FOR n=1 TO 2000
40 PRINT CHR$(93);
50 NEXT
60 GOTO 60
```

KÄSKY: SYMBOL-käskyä käytetään merkkien ulkonäön uudelleen määrittämiseen. Ensin on kuitenkin määriteltävä sen merkin numero, jonka jälkeen uudelleen määriteltävä merkki on ASCII-merkistössä. Tämä tapahtuu SYMBOL AFTER-käskyllä. Yllä olevassa esimerkissä rivi 10 SYMBOL AFTER 90 tarkoittaa sitä, että uudelleen määriteltävä merkki (= merkin numero) on merkin nro 90 jälkeen (tai merkki nro 90). 'Merkin numero' valitaan siis ASCII-merkkivalikoimasta tai muusta CPC 464:n vakiomerkkivalikoimasta. Varsinainen merkin määrittely tapahtuu esimerkkiohjelmassa rivillä 20, jossa SYMBOL 93 määrittelee sen merkin (merkki nro 93 eli]), jonka ulkonäkö halutaan muuttaa. Loput rivin 20 kahdeksan lukua (tässä heksadesimaalimuodossa, mutta voivat olla myös desimaalimuodossa) määrittelevät uuden merkin ulkomuodon 8 x 8 kuva-alkiomatriisilla. Kun tietokone muuttaa luvut binäärimuotoon (8-bittisiksi luvuiksi), tarkoittaa 0 sitä, että nollan kohdalla on käytettävä paperin väriä ja 1 puolestaan sitä, että kuva-alkio on asetettava senhetkisen musteen väriseksi. Asiaa käsitellään myös liitteissä II ja III. Yllä oleva esimerkkiohjelma tuottaa merkkisolun (8 x 8 pistettä) läpi kulkevan vinoviivan, joka siis löytyy]-näppäimen alta.

Aiheeseen liittyvät avainsanat: SYMBOL AFTER

SYMBOL AFTER

SYMBOL AFTER ´kokonaislukulauseke´

SYMBOL AFTER 90

KÄSKY: Käyttäjä voi itse määritellä suurimman osan CPC 464:n merkeistä uudelleen. Uudelleen määriteltävien merkkien lukumäärä määritellään SYMBOL AFTER-käskyllä. Oletusarvo on 240 (SYMBOL AFTER 240), jolloin käyttäjä voi määritellä uudelleen 16 merkkiä (merkit 240 - 255). Jos ´kokonaislukulauseke´ on 32, niin käyttäjä voi määritellä kaikki merkit väliltä 32 - 255.

Aina kun SYMBOL AFTER-käskyä käytetään, palautuvat kaikki käyttäjän määrittelemät merkit takaisin oletusarvoihinsa.

Aiheeseen liittyvät avainsanat: SYMBOL

TAG

TAG [#´reittilauseke´]

```
10 MODE 2
11 BORDER 9
14 INK 0,12
15 INK 1,0
20 FOR n=1 TO 100
30 MOVE 200+n,320+n
40 TAG
50 IF n<70 GOTO 60 ELSE 70
60 PRINT "Terve!";:GOTO 80
70 PRINT " Hei hei!";
80 NEXT
90 GOTO 20
```

KÄSKY: Määritellylle tulostusreitille lähetetty teksti voidaan ohjata uudelleen kirjoitettavaksi grafiikkakohdistimen paikkaan. Tämä suo mahdollisuuden sekoittaa tekstiä ja symboleja grafiikan kanssa. ´Reittilausekkeen´ oletusarvo on 0. Merkkisolun vasen yläkulma "kiinnitetään" grafiikkakohdistimeen, ja kirjoittumattomat ohjausmerkit (esim. rivinvaihtomerkki) tulostuvat ruutuun, ellei PRINT-lauseketta päätetä puolipisteellä ; .

Aiheeseen liittyvät avainsanat: TAGOFF

TAGOFF

TAGOFF [#´reittilauseke´]

TAGOFF #0

KÄSKY: TAGOFF-käsky peruuttaa TAG-käskyn vaikutuksen määritellyltä reitiltä ja lähettää tekstin edelliseen tekstikohdistimen paikkaan kohdassa, jossa "kiinnitys" TAG-käskyllä tapahtui.

Aiheeseen liittyvät avainsanat: TAG

TAN

TAN('numerolauseke')

PRINT TAN(45)

FUNKTIO: TAN-funktio laskee kohdassa 'numerolauseke' määritellyn kulman tangentin. Numerolausekkeen on oltava väliltä -200000...+200000. Mittayksikkönä käytetään oletusarvona radiaaneja, joka voidaan muuttaa DEG-käskyllä asteiksi.

Aiheeseen liittyvät avainsanat: COS, SIN, ATN, DEG, RAD

TEST

TEST('x-koordinaatti','y-koordinaatti')

PRINT TEST(300,300)

FUNKTIO: TEST-funktiolla saadaan selville senhetkinen musteen arvo x- ja y-koordinaateilla määritellyssä grafiikkakohdistimen paikassa.

Aiheeseen liittyvät avainsanat: TESTR, MOVE, MOVER, PLOT, PLOTR, DRAW, DRAWR

TESTR

TESTR('x-siirtymä','y-siirtymä')

TESTR(5,5)

FUNKTIO: TESTR-funktiolla saadaan selville senhetkinen musteen väri määritellyssä grafiikkakohdistimen paikassa sen jälkeen kun grafiikkakohdistimen absoluuttisiin koordinaattiarvoihin on lisätty sulkeissa määritellyt siirtymät.

Aiheeseen liittyvät avainsanat: TEST, MOVE, MOVER, PLOT, PLOTR, DRAW, DRAWR

TIME

TIME

```
10 ALKUHETKI = INT(TIME/300)
20 SEKUNTIKELLO=((TIME/300)-ALKUHETKI)
30 PRINT SEKUNTIKELLO;
40 GOTO 20
```

FUNKTIO: TIME-funktiolla saadaan selville, kuinka pitkä aika on kulunut siitä kun virta kytkettiin tietokoneeseen. Tähän aikaan ei ole laskettu mukaan jaksoja, jolloin kasetille on tallennettu tai kasetilta on luettu tietoja. Aikayksikkönä käytetään 1/300 sekuntia.

TRON
TROFF

TRON
TROFF

TRON

KÄSKYJÄ: CPC 464:n BASIC sisältää mahdollisuuden jäljittää ohjelman suoritusta siten, että kunkin rivin numero ilmoitetaan hakasulkeissa [ ] juuri ennen rivin suorittamista. TRON-käsky kytkee jäljityksen toimintaan ja TROFF-käsky katkaisee sen pois toiminnasta.

Aiheeseen liittyvät avainsanat: RUN

UNT

UNT(´osoitelauseke´)

PRINT UNT(&FF66)

FUNKTIO: Muuntaa etumerkittömän 16-bittisen kokonaisluvun väliltä 0...65535. Antaa vastaukseksi kokonaisluvun väliltä -32768...+32767.

Aiheeseen liittyvät avainsanat: INT, FIX, CINT, ROUND

UPPER$

UPPER$(´merkkijonolauseke´)

PRINT UPPER$("amstrad")
AMSTRAD

FUNKTIO: Muuntaa minkä hyvänsä kohdassa ´merkkijonolauseke´ olevan pienen kirjaimen isoksi kirjaimeksi.

Aiheeseen liittyvät avainsanat: LOWER$

VAL

VAL(´merkkijonolauseke´)

10 A$="9 on onnennumeroni"
20 PRINT VAL(A$)

FUNKTIO: VAL-funktio erottaa ´numerolausekkeen´´merkkijonolausekkeen´alusta. VAL-funktio on vastakohta STR$-funktiolle.

Aiheeseen liittyvät avainsanat: STR$

VPOS

VPOS(#´reittilauseke´)

PRINT VPOS(#0)

FUNKTIO: VPOS-funktiolla saadaan selville ´reittilausekkeen´ tekstikohdistimen paikka pystysuunnassa. Reittilauseke on aina määriteltävä.

Aiheeseen liittyvät avainsanat: POS

WAIT

WAIT ´portin numero´,´maski´[,´inversio´]

WAIT &FF34,20,25

KÄSKY: WAIT-käsky keskeyttää toiminnan, kunnes määritellyltä I/O-portilta saadaan tietty arvo, joka on väliltä 0...255. BASIC suorittaa silmukkaa sillä aikaa kun se lukee I/O-portin tilaa. Luetulle arvolle suoritetaan looginen operaatio POISSULKEVA TAI (EXCLUSIVE OR) käytettäessä ´inversiota´ ja operaatio JA (AND) käytettäessä ´maskia´ kunnes saatu tulos poikkeaa nollasta. BASIC jumiutuu WAIT-silmukkaan, ellei vaadittu ehto täyty. Jos näppäilet yllä olevan esimerkkirivin, sinun on nollattava tietokone aivan kokonaan päästäksesi ulos silmukasta.

Aiheeseen liittyvät avainsanat: INP, OUT

WEND

WEND

```
10 MODE 1:REM KELLONAJAN PERUSOHJELMA
20 INPUT "Anna tämänhetkinen kellonaika
eli tunnit minuutit ja sekunnit (t,m,s)"
;tu,mi,se
30 CLS:alku = INT(TIME/300)
40 WHILE tu<25
50 WHILE mi<60
60 WHILE sek<60
70 sek=(INT(TIME/300)-alku)+se
80 LOCATE 70,4
90 PRINT #0,USING "## ";tu,mi,sek
100 WEND
110 sek=0
115 se=0
120 mi=mi+1
130 GOTO 30
140 WEND
150 mi=0
160 tu=tu+1
170 WEND
180 tu=1
190 GOTO 40
```

KÄSKY: WHILE/WEND-silmukka suorittaa ohjelman runkoa yhä uudelleen ja uudelleen, kunnes määritelty ehto täyttyy. Yllä oleva ohjelma on esimerkki siitä, kuinka kätevästi WHILE/WEND-silmukoita voi käyttää vaikkapa kellonaika-ohjelmissa. Voit itse lisätä ja täydentää ohjelmaa tai muuttaa sen mieleiseksesi. WEND-käsky päättää siis WHILE-silmukan.

Aiheeseen liittyvät avainsanat: WHILE

WHILE

WHILE ´looginen lauseke´

WHILE DAY < 0

Katso edellä olevaa esimerkkiohjelmaa.

KÄSKY: WHILE/WEND-silmukka toistaa ohjelman runkoa yhä uudelleen ja uudelleen, kunnes määritelty ehto täyttyy. WHILE-käsky määrittelee silmukan pään. Samalla se määrittelee ehdon jonka on täytyttävä, ennen kuin silmukka päättyy. WHILE-silmukka päätetään WEND-käskyllä.

Aiheeseen liittyvät avainsanat: WEND

WIDTH

WIDTH ´kokonaislukulauseke´

WIDTH 86

KÄSKY: WIDTH-käsky ilmoittaa BASIC-ohjelmalle, kuinka leveä kirjoittimen tulostusleveys on merkeissä mitattuna. Tämän tiedon avulla BASIC-ohjelma pystyy upottamaan vaununpalautukset tarvittaviin kohtiin tulostuksen aikana.

Aiheeseen liittyvät avainsanat: PRINT, POS

WINDOW

WINDOW [#´reittilauseke´,]´vasen´,´oikea´,´ylin´,´alin´

```
10 MODE 1
20 BORDER 6
30 WINDOW 10,30,7,18
40 PAPER 2:PEN 3
50 CLS
60 PRINT CHR$(143);CHR$(242);"TÄMÄ ON PA
IKKA"
70 PRINT "1,1 TEKSTI-IKKUNASSA"
80 GOTO 80
```

KÄSKY: WINDOW-käskyllä asetetaan teksti-ikkuna tietylle kuvaruudun tulostus-reitille.

Aiheeseen liittyvät avainsanat: ORIGIN

WINDOW SWAP

WINDOW SWAP ´reittilauseke´,´reittilauseke´

WINDOW SWAP 0,2

KÄSKY: Vaihtaa teksti-ikkunoita. Esimerkiksi tulostusreitille #0 lähetytyt BASIC-viestit voidaan vaihtaa toiseen ikkunaan haluttaessa korostaa ohjelman kehitykseen ja toimintaa liittyviä seikkoja.

Aiheeseen liittyvät avainsanat: WINDOW, PEN, PAPER, TAG

WRITE

WRITE [#´reittilauseke´,][´kirjoitettava luettelo´]

```
WRITE #2,"HALKOJA",4,5
"HALKOJA",4,5
```

KÄSKY: WRITE-käsky tulostaa lausekejoukon arvot määritellylle tulostusreitille erottaen ne pilkuilla ja sijoittaen merkkijonot lainausmerkkien väliin.

Aiheeseen liittyvät avainsanat: PRINT

XPOS

XPOS

PRINT XPOS

FUNKTIO: XPOS-funktio ilmoittaa grafiikkakohdistimen paikan vaakasuunnassa.

Aiheeseen liittyvät avainsanat: YPOS, MOVE, MOVER, ORIGIN

YPOS

YPOS

PRINT YPOS

FUNKTIO: YPOS-funktio ilmoittaa grafiikkakohdistimen paikan pystysuunnassa.

Aiheeseen liittyvät avainsanat: XPOS, MOVE, MOVER, ORIGIN

ZONE

ZONE ´kokonaislukulauseke´

```
10 PRINT 1,2,3
20 ZONE 19
30 PRINT 4,5,6
```

KÄSKY: Muuttaa PRINT-käskyn yhteydessä käytettävän tulostusvyöhykkeen leveyden, jonka oletusarvo on 13. Uudeksi arvoksi voidaan valita mikä hyvänsä merkkimäärä väliltä 1...255. Tulostusvyöhyke palautuu oletusarvoonsa NEW-, LOAD-, CHAIN- ja RUN "´tiedoston nimi´"-käskyillä.

Aiheeseen liittyvät avainsanat: PRINT, WIDTH

LUKU 9: LISÄTIETOJA OHJELMOINNISTA

Tässä luvussa käsiteltäviä aiheita ovat:

- Tekstin sallitut kohdat
- Ohjausmerkit
- Koneen käyttöjärjestelmä
- Keskeytysrakenteet

9.1 KOHDISTIMEN PAIKAT JA OHJAUSKOODIT

Tekstikohdistin voidaan sijoittaa senhetkisen ikkunan ulkopuolelle lukuisissa sovellutusohjelmissa. Toisaalta lukuisat toiminnot pakoittavat kohdistimen sallittuun paikkaan ennen toimintojen suorittamista. Tällaisia toimintoja ovat:

- merkin kirjoitus
- kohdistinneliön kirjoitus
- seuraavilla sivuilla olevassa luettelossa tähdellä * merkittyjen ohjauskoodien määräämän toiminnon toteutus.

Kohdistin pakoitetaan sallituun paikkaan seuraavasti:

a. Jos kohdistin on oikean reunan oikealla puolella, niin kohdistin siirretään seuraavan alemman rivin vasemmanpuolimmaiseen sarakkeeseen.

b. Jos kohdistin on vasemman reunan vasemmalla puolella, niin kohdistin siirretään seuraavan ylemmän rivin oikeanpuolimmaiseen sarakkeeseen.

c. Jos kohdistin on yläreunan yläpuolella, niin ikkunaa vieritetään alaspäin yksi rivi ja kohdistin asetetaan ikkunan ylimmälle riville.

d. Jos kohdistin on alareunan alapuolella, niin ikkunaa vieritetään ylöspäin yksi rivi ja kohdistin asetetaan ikkunan alimmalle riville.

Testit ja operaatiot tehdään määritellyssä järjestyksessä. Luvattomat kohdistimen paikat voivat olla nollia tai negatiivisia, jolloin kohdistin on ikkunan vasemmalla puolella tai yläpuolella.

Merkkien nro 0...31 (katso liitettä III) lähettäminen tekstiruutuun ei aiheuta merkin tulostumista ruutuun (ja se saattaa aiheuttaa tietokoneen käyttöjärjestelmän lukittumisen, jos ohjausmerkkejä käytetään miten sattuu), vaan merkit tulkitaan OHJAUSKOODEIKSI. Eräät koodeista muuttavat koodin jälkeen olevan yhden tai useamman merkin merkitystä, jotka ovat koodin parametrejä.

Grafiikkaruutuun lähetettävät ohjausmerkit tulostuvat (katso avainsanan TAG kuvausta luvusta 8), ja merkin &07 "BEL" kaltaiset merkit tulostavat näkyviin niiden toimintaan liittyvän symbolin, mikäli ohjausmerkki näppäillään näppäimistöltä (esim. painamalla CTRL- ja G-näppäimiä). Sitä vastoin ne suorittavat pelkästään niihin kuuluvan toiminnan, mikäli niitä osoitetaan käyttäen käskymuotoa PRINT CHR$(&07).

Tähtimerkillä * merkityt koodit pakoittavat kohdistimen sallittuun paikkaan teksti-ikkunassa, ennen kuin ohjauskoodin määräämä toiminto suoritetaan. Kohdistin voi silti jäädä laittomaan paikkaan. Koodeja ja niiden merkitystä on kuvattu ensin niiden heksadesimaaliarvolla (&XX) ja sitten vastaavalla desimaaliarvolla.

Ohjausmerkein toteutettavat lisäkomennot, joita ei yleensä saa käyttöön CTRL-näppäimen avulla:

Arvo		Nimi	Parametri	Merkitys
&00	0	NUL		Ei vaikutusta. Jätetään huomioimatta.
&01	1	SOH	0...255	Tulostaa parametriarvolla määritellyn symbolin. Tämä suo mahdollisuuden tulostaa näyttöön symbolit väliltä 0...31.
&02	2	STX		Sammuttaa tekstikohdistimen.
&03	3	ETX		Sytyttää tekstikohdistimen. Huom. BASIC käyttää syrjäyttävää kohdistimen estoa, joka vapautetaan vain sen odottaessa näppäilyjä näppäimistöltä.
&04	4	EOT	0...2	Asettaa näyttömuodon. Valittu parametri MOD 4. Vastaa MODE-käskyä.
&05	5	ENQ	0...255	Lähettää parametrillä määritellyn merkin grafiikkakohdistimeen.
&06	6	ACK		Estää tekstiruudun. (Katso &15, NAK, seuraava sivu).
&07	7	BEL		Äänimerkki. Huom. tämä tyhjentää äänijonot.
&08	8 *	BS		Siirtää kohdistinta taaksepäin yhden merkin.
&09	9 *	TAB		Siirtää kohdistinta eteenpäin yhden merkin.
&0A	10 *	LF		Siirtää kohdistinta alaspäin yhden rivin.
&0B	11 *	VT		Siirtää kohdistinta ylöspäin yhden rivin.
&0C	12	FF		Tyhjentää teksi-ikkunan ja siirtää kohdistimen vasempaan ylänurkkaan. Vastaa CLS-käskyä.
&0D	13 *	CR		Siirtää kohdistimen ikkunan vasempaan reunaan senhetkisellä rivillä.
&0E	14	SO	0...15	Asettaa paperin musteen. Valittu parametri MOD 16. Vastaa PAPER-käskyä.
&0F	15	SI	0...15	Asetaa kynän musteen. Valittu parametri MOD 16. Vastaa PEN-käskyä.
&10	16 *	DLE		Poistaa senhetkisen merkin. Täyttää merkkisolun senhetkisen paperin musteella.
&11	17 *	DC1		Tyhjentää ikkunan vasemmasta reunasta senhetkiseen merkkipaikkaan asti kyseinen merkkipaikka mukaan luettuna. Täyttää asiaankuuluvat merkkisolut senhetkisellä paperin musteella.

Arvo		**Nimi**	**Parametri**	**Merkitys**
&12	18 *	DC2		Tyhjentää ikkunan oikeaan reunaan senhetkisestä merkkipaikasta kyseinen merkkipaikka mukaan luettuna. Täyttää asiaankuuluvat merkkisolut senhetkisellä paperin musteella.
&13	19 *	DC3		Tyhjentää ikkunan alusta senhetkiseen merkkipaikkaan kyseinen merkkipaikka mukaan luettuna. Täyttää asiaankuuluvat merkkisolut senhetkisellä paperin musteella.
&14	20 *	DC4		Tyhjentää ikkunan loppuun senhetkisestä merkkipaikasta kyseinen merkkipaikka mukaan luettuna. Täyttää asiaankuuluvat merkkisolut senhetkisellä paperin musteella.
&15	21	NAK		Sammuttaa tekstiruudun. Ruutu ei reagoi mihinkään siihen lähetettyyn tietoon ennen kuin ACK(&06 6) on lähetetty.
&16	22	SYN	0...1	Parametri MOD2 0 estää läpinäkyvän näyttötavan ja 1 sallii sen.
&17	23	ETB	0...3	Parametri MOD4 0 asettaa normaalin grafiikkamustetilan 1 XOR 2 AND 3 OR
&18	24	CAN		Vaihtaa kynän ja paperin musteet.
&19	25	EM	0...255 0...255 0...255 0...255 0...255 0...255 0...255 0...255 0...255	Asettaa matriisin käyttäjän määriteltävissä oleville merkeille. Vastaa SYMBOL-käskyä. Vaatii 9 parametriä, joista ensimmäinen määrittelee sen merkin, jonka matriisi asetetaan. Seuraavat kahdeksan määrittelevät matriisin: ensimmäisen tavun eniten merkitsevä bitti vastaa merkkisolun vasemmassa yläkulmassa olevaa kuva-alkiota ja viimeisen tavun vähiten merkitsevä bitti vastaa merkkisolun oikeassa alakulmassa olevaa kuva-alkiota.
&1A	26	SUB	1...80 1...80 1...25 1...25	Asettaa ikkunan. Vastaa WINDOW-käskyä. Kaksi ensimmäistä parametriä määrittelevät ikkunan vasemman ja oikean reunan; pienempi arvo valitaan vasemmaksi ja suurempi oikeaksi reunaksi. Kaksi seuraavaa parametriä määrittelevät ikkunan ylä- ja alareunat; pienempi arvo valitaan yläreunaksi ja suurempi alareunaksi.

Arvo		Nimi	Parametri	Merkitys
&1B	27	ESC		Ei vaikutusta. Jätetään huomioimatta.
&1C	28	FS	0...15 0...31 0...31	Asettaa musteen väriparin väriseksi. Vastaa INK-käskyä. Ensimmäinen parametri (MOD 16) määrittelee musteen ja kaksi seuraavaa (MOD 32) halutut värit.
&1D	29	GS	0...31 0...31	Asettaa reuna-alueen väriparin väriseksi. Vastaa BORDER-käskyä. Kaksi parametriä (MOD 32) määrittelevät halutut kaksi väriä.
&1E	30	RS		Siirtää kohdistimen ikkunan vasempaan yläkulmaan.
&1F	31	US	1...80 1...25	Siirtää kohdistimen senhetkisen ikkunan määriteltyyn paikkaan. Vastaa LOCATE-käskyä. Ensimmäinen parametri määrittelee sarakkeen ja toinen rivin.

9.2 KONEEN KÄYTTÖJÄRJESTELMÄ

CPC 464:n sisäisestä toiminnasta vastaa varsin monipuolinen tosiaikainen käyttöjärjestelmä. Käyttöjärjestelmä ohjaa tietokoneen kautta kulkevaa liikennettä eri syöttö- ja tulostuspisteiden välillä.

Käyttöjärjestelmän ensisijaisena tehtävänä on toimia laitteiston ja BASIC-tulkin välillä. Esimerkkinä mainittakoon vaikkapa värien vilkutustoiminto, jossa parametrit saadaan BASICista ja käyttöjärjestelmä huolehtii itse tehtävän suorittamisesta. Yhdellä osalla siis määritellään mitä halutaan, ja toinen osa määrittelee tapahtumien ajoituksen.

Tietokoneen käyttöjärjestelmää nimitetään yleensä kiinteäksi laiteohjelmistoksi (firmware), ja se koostuu konekielisistä rutiineista, joita kutsutaan BASICin korkean tason käskyillä.

Vaikka CPC 464:n toimintaa onkin (lähes) mahdotonta "jumiuttaa" BASICilla - muuten kuin käyttämällä ON BREAK GOSUB-käskyä - on varsin helppoa saada aikaan ongelmia sorkkimalla koneen käyttöjärjestelmää. Tällainen lukitus saadaan puretuksi vain täydellisellä tietokoneen nollauksella, jolloin kaikki muistissa olevat ohjelmat, tiedot yms. tuhoutuvat.

Jos sinua kuitenkin kiinnostaa tutkia tietokoneen muistialuetta POKE-käskyllä ja kutsua alirutiineja CALL-käskyllä, tallenna ohjelmasi ja listauksesi ennen muistin penkomista, sillä muuten siinä saattaa käydä kalpaten! CPC 464:n käyttöjärjestelmän kiinteää laiteohjelmistoa kuvataan tarkemmin pitemmälle ehtineiden käyttäjien käsikirjassa eikä se kuulu tämän perusteita käsittelevän käsikirjan aihepiiriin.

9.3 KESKEYTYKSET

CPC 464:ssä on käytetty sangen laajalti Z80-prosessorin keskeytysmahdollisuuksia. Näin on saatu aikaan käyttöjärjestelmä, johon kuuluu lukuisia toimintoja, joissa suoritetaan useampia tehtäviä samanaikaisesti. Näitä on kuvattu esimerkein luvussa 8 avainsanojen AFTER ja EVERY kohdalla. CPC 464:n tapahtuma-ajastimien etuoikeusjärjestys on seuraava:

Break (kaksi perättäistä ESC-näppäimen painallusta)
Ajastin 3
Ajastin 2 (ja kolme äänikanavajonoa)
Ajastin 1
Ajastin 0

Keskeytyksiä voi laittaa ohjelmiin sen jälkeen kun on huomioitu kaikki seuraukset, joita keskeytyshetkellä olevien muuttujien senhetkiset tilat aiheuttavat. Pitää yrittää välttää, ettei pääohjelmassa olevan muuttujan tila vaikuta haitallisesti itse keskeyttävään aliohjelmaan.

Äänijonoilla on toisistaan riippumattomat samanarvoiset keskeytykset. Äänikeskeytyksen käynnistyttyä sitä ei voi keskeyttää mikään muu äänijono, jolloin äänikäskyt voivat jakaa keskenään muuttujien arvoja niiden vaikuttamatta haitallisesti edellisessä kappaleessa mainitulla tavalla.

Kun äänijonon keskeytys on sallittu, se keskeytyy välittömästi jos kyseisen kanavan äänijono ei ole täysi. Se siis keskeytyy seuraavan äänen käynnistyessä, jos jonossa on vielä tilaa. Keskeytystoiminto estää tapahtuman, joten aliohjelman täytyy sallia se uudelleen, mikäli keskeytyksiä vielä tarvitaan. Mikä hyvänsä tapahtuma estetään myös silloin, jos yritetään muodostaa ääni tai testata jonon tila.

ESC-näppäimen kaksi perättäistä painallusta on siis etuoikeudeltaan kaikkein tärkein keskeytys. Tällä on haluttu varmistaa se, että BASIC-ohjelman toiminta voidaan pysäyttää ohjelmaa tuhoamatta. Edellytyksenä tälle on tietenkin se, ettei ohjelmaa ole suojattu. Suojaustapoja on useampia, ja suojattua ohjelmaa ei voi yleensä pysäyttää ESC-näppäimellä.

9.4 AMSOFTIN SYMBOLISEN KONEKIELEN KÄÄNTÄJÄ (ASSEMBLER)

Jos ohjelmoinnissa halutaan käyttää laajalti konekieltä, on välttämätöntä käyttää assembler-ohjelmaa. AMSOFTin assembler-ohjelma käsittää uudelleensijoitettavan Z80-assemblerin, johon kuuluu editori, disassembleri ja monitori.

LUKU 10: KESKEYTYSMAHDOLLISUUDET

Tässä luvussa käsiteltäviä asioita ovat:

- AFTER
- EVERY
- REMAIN
- Pääkello

Olet jo ehkä pannutkin merkille, että CPC 464:n ohjelmiston tärkein piirre on sen ainutlaatuinen mahdollisuus käsitellä BASIC-käskyillä ohjattavia keskeytyksiä. Tämä merkitsee sitä, että AMSTRADin BASIC pystyy suorittamaan lukuisia samanaikaisia mutta silti erillisiä operaatioita saman ohjelman puitteissa. Joissakin yhteyksissä tästä käytetään nimitystä "monen tehtävän samanaikainen suoritus", ja se on toteutettu CPC 464:ssä käyttämällä uusia käskyjä AFTER ja EVERY.

Keskeytysmahdollisuudet tulevat selvästi esille mm. siinä, miten CPC 464 pystyy käsittelemään äänijonoja ja niiden tahdistuksia.

Kaikki ajoitustapahtumat riippuvat aina järjestelmän pääkellosta. CPC 464:ssä se on kvartsikideohjattu ajoitusjärjestelmä, joka huolehtii tietokoneessa olevien tapahtumien ajoituksesta ja tahdistuksesta. Tällaisia tapahtumia ovat esim. näytön "skannaus" ja prosessorin ajastus. Jos jokin laitteiston toiminto on tavalla tai toisella riippuvainen ajasta, löytyy senkin toiminnan taustalta aina järjestelmän pääkello.

AMSTRADin BASICissa keskeytystoimintoja päästään siis hyödyntämään AFTER- ja EVERY-käskyillä, joita kuvataan tarkemmin seuraavissa kappaleissa.

10.1 AFTER

Kuten jo edellä mainittiin, on CPC 464:ssä reaaliaikainen kello. AFTER-käskyä voidaan käyttää BASIC-ohjelmissa siten, että aliohjelmia voidaan kutsua AFTER-käskyllä määritellyn ajan kuluttua. Tähän on käytettävissä neljä viiveajastinta, ja kuhunkin niistä voidaan liittää oma aliohjelmansa. AFTER-käsky on muotoa:

AFTER ´kokonaislukulauseke´[,´kokonaislukulauseke´] GOSUB ´rivinumero´

Ensimmäinen ´kokonaislukulauseke´ määrittelee ajan, jonka jälkeen aliohjelmaa kutsutaan. Aikayksikkönä käytetään 1/50 sekuntia.

Toinen ´kokonaislukulauseke´ määrittelee yhden neljästä käytettävissä olevasta viiveajastimesta. Lausekkeen arvon on oltava väliltä 0...3. Oletusarvo on 0, ellei sitä erikseen määritellä.

Kun kokonaislukulausekkeella määritelty aika on kulunut, aliohjelmaa kutsutaan automaattisesti aivan kuin kyseisessä ohjelman kohdassa olisi ollut GOSUB-käsky. Kun aliohjelma päättyy normaalilla RETURN-käskyllä, jatkuu pääohjelman suoritus siitä kohdasta, jossa se keskeytettiin.

Keskeytysajastimilla on erilaiset etuoikeusarvot. Ajastimella 3 on korkein ja ajastimella 0 on pienin etuoikeusarvo.

AFTER-käskyjä voidaan laittaa mihin ohjelman kohtaan hyvänsä, jolloin kyseiseen viiveajastimeen liittyvä aliohjelma ja aika palautuvat uudelleen käyttöön. Viiveajastimet ovat samat kuin EVERY-käskyissä käytettävät viiveajastimetkin, joten AFTER-käsky syrjäyttää minkä hyvänsä samalle ajastimelle määritellyn aikaisemman EVERY-käskyn, ja päinvastoin.

```
10 MODE 1:X=0
20 AFTER 45 GOSUB 100
30 AFTER 100,1 GOSUB 200
40 PRINT "AMSOFT"
50 WHILE X 100
60 LOCATE #1,30,1:PRINT #1,X:X=X+1
70 WEND
80 END
100 PRINT "oheislaitteet"
110 RETURN
200 PRINT "ja ohjelmat"
210 RETURN
```

Huomaa, että ohjelmassa on käytetty kahta tulostusreittiä (ikkunaa), jolloin riveillä 50 - 80 oleva "pääohjelma" voi tulostaa omaan kohdistimen paikkaansa riippumatta siitä, mitä kohdistimen paikkaa keskeyttävät aliohjelmat käyttävät.

Kun AFTER-käskyllä määritelty aika on kulunut, ohjelma haarautuu aliohjelmaan ja suorittaa siellä määritellyn tehtävän.

10.2 EVERY

BASIC-ohjelma saadaan kutsumaan aliohjelmia säännöllisin välein. Tämä tapahtuu EVERY-käskyllä. Käytettävissä on neljä viiveajastinta, ja kuhunkin niistä voi liittyä oma aliohjelmansa. EVERY-käskyn muoto on seuraava:

EVERY ´kokonaislukulauseke´[,´kokonaislukulauseke´] GOSUB ´rivinumero´

Ensimmäinen ´kokonaislukulauseke´ määrittelee, kuinka kauan odotetaan kunkin aliohjelmakutsun välillä. Aikayksikkönä käytetään 1/50 sekuntia.

Toinen ´kokonaislukulauseke´ määrittelee yhden neljästä käytettävissä olevasta viiveajastimesta. Lausekkeen arvon on oltava väliltä 0...3. Oletusarvo on 0, ellei sitä erikseen määritellä.

Kun kokonaislukulausekkeella määritelty aika on kulunut, aliohjelmaa kutsutaan automaattisesti aivan kuin kyseisessä ohjelman kohdassa olisi ollut GOSUB-käsky. Kun aliohjelma päättyy normaalilla RETURN-käskyllä, jatkuu pääohjelman suoritus siitä kohdasta, jossa se keskeytettiin.

Keskeytysajastimilla on erilaiset etuoikeusarvot. Ajastimella 3 on korkein ja ajastimella 0 on pienin etuoikeusarvo. Välittömästi määritellyn ajan kuluttua aikalukema palautetaan alkuarvoonsa ja ajastin alkaa laskea seuraavaan aliohjelmakutsuun käytettävää aikaa.

EVERY-käskyjä voidaan laittaa mihin ohjelman kohtaan hyvänsä, jolloin kyseiseen viiveajastimeen liittyvä aliohjelma ja aika palautuvat uudelleen käyttöön. Viiveajastimet ovat samat kuin AFTER-käskyissä käytettävät viiveajastimetkin, joten EVERY-käsky syrjäyttää minkä hyvänsä samalle ajastimelle määritellyn aikaisemman AFTER-käskyn, ja päinvastoin.

```
10 MODE 1:X=0
20 P100=0:EVERY 10 GOSUB 100
30 P200=0:EVERY 12,1 GOSUB 200
40 PRINT "AMSOFT"
50 WHILE X<200
60 LOCATE #1,30,1:PRINT #1,X:X=X+1
70 WEND
80 LOCATE 1,20:END
100 DI:PEN P100:LOCATE 1,2: PRINT "oheis
laitteet":EI
105 IF P100=0 THEN P100=1 ELSE P100=0
110 RETURN
200 PEN P200:LOCATE 1,3: PRINT "ja ohjel
mat"
205 IF P200=2 THEN P200=3 ELSE P200=2
210 RETURN
```

Huomaa ohjelmassa käytetyt DI- ja EI-käskyt, jotka estävät (DI) ja sallivat (EI) ajastinkeskeytykset sillä aikaa kun niiden välisiä käskyjä suoritetaan. Tällä on se vaikutus, että ajastimen 1 keskeytystä (jolla on korkeampi etuoikeusarvo) viivästetään niin paljon, ettei se ehdi tapahtua lainkaan ajastimen 0 keskeytyksen (rivit 100 - 110) käsittelyn aikana, joten se ei myöskään ehdi sekoittaa PEN- tai LOCATE-asetuksia ennen PRINT-käskyä.

10.3 REMAIN

REMAIN-funktiolla saadaan selville järjestelmän neljän viiveajastimen jäljellä oleva "saldo". Se estää ajastimen toiminnan ja antaa vastaukseksi 0, jos ajastimen toiminta oli jo estetty. REMAIN-funktiota käytetään seuraavassa muodossa:

REMAIN (´kokonaislukulauseke´)

LIITE I: TIETOKONEEN OMINAISUUKSIA

Tietokoneiden parissa harrastustaan aloittelevat henkilöt selvittävät yleensä tietokoneista muutamia seikkoja, joita vertailemalla he luovat itselleen käsityksen eri tietokoneiden mahdollisuuksista ja rajoituksista. Tässä luvussa selvitellään juuri tietokoneiden tärkeimpiä ominaisuuksia, jonka jälkeen käydään läpi keskeisintä tietokonesanastoa.

Vaikka ainoa syy CPC 464:n hankkimiseksi olisikin ollut halu päästä hyödyntämään laitteistollasi hienoja tietokonepelejä, saatat silti ihmetellä joitakin laitteistoon liittyviä ominaisuuksia.

Laitteistoksi kutsutaan sitä kovaa ja kosketeltavaa kalustoa, jonka voit ottaa kainaloosi ja kantaa toiseen paikkaan. Laitteistoon kuuluu siis varsinainen tietokoneyksikkö, näppäimistö, monitori, liitosjohdot, jne. Oikeastaan laitteistoon kuuluu kaikki muu mikä ei ole erikseen määriteltävissä ohjelmistoksi, johon puolestaan kuuluvat ohjelmat, käsikirjat sekä kasettipohjaiset tiedot.

Tietyt tietokoneen käyttäytymiseen vaikuttavat seikat ovat peräisin laitteiston ominaisuuksista. Tällaisia ovat esim. television (tai monitorin) värinäyttö. Tästä eteenpäin kaikki riippuu ohjelmistosta, eli kuinka hyvin ohjelmisto pystyy käyttämään hyväkseen laitteiston kykyä tuottaa haluttuja merkkejä ja kuvioita kuvaruutuun.

Laitteisto huolehtii oikeastaan vain siitä, että se ohjaa elektronisuihkua TV-kuvaputken kuvaruudun sisäpinnalla olevalla elektroluminesenssipinnalla (joka siis muodostaa näkyvää valoa elektronisuihkun osuessa siihen). Ohjelmisto on sitten se, jonka ansiosta kuvaruutuun saadaan näkyviin jotain järkevää, koska ohjelmisto ohjaa laitteistoa ilmoittaen milloin mitäkin on tehtävä. Ohjelmisto huolehtii ajoituksesta, ohjauksesta ja asioiden oikeasta suoritusjärjestyksestä. Juuri ohjelmiston ansiota on, että avaruusalus saadaan laukaistuksi tarkasti oikealla hetkellä. Ohjelmisto huolehtii myös vähemmän näyttävistä tapahtumista, kuten kirjaimen tulostamisesta kuvaruutuun painettaessa näppäimistön vastaavaa näppäintä.

TIETOKONEEN KESKEISIÄ OMINAISUUKSIA

Pelkällä laitteistolla ilman ohjelmistoa ei tee yhtään mitään. Yhtä lailla ohjelmisto ilman laitteistoa on täysin arvotonta, joten tietokoneella alkaa olla arvoa vasta sitten, kun laitteisto ja ohjelmisto alkavat toimia yhdessä tiettyjen tehtävien suorittamiseksi. Tarkastelemme seuraavaksi eräitä keskeisiä asioita, joiden perusteella voidaan arvioida sekä laitteiston että ohjelmiston suorituskykyä. Tällaisia henkilökohtaisten tietokoneiden yleisesti hyväksyttyjä vertailukohtia ovat:

1. Kuvan erottelukyky eli pienin osoitettavissa oleva yksittäinen kuva-alkio

Kuvan erottelukyky riippuu useammastakin seikasta, joita ovat mm. kullakin hetkellä ohjelmoijan käytettävissä olevien eri värien määrä sekä kuvaruudussa näkyvien toisistaan selvästi erottuvien pienimpien kuva-alkioiden (pixel) määrä. Vertailun perusteena voidaan käyttää myös yhdellä kerralla kuvaruutuun mahtuvien tekstimerkkien lukumäärää.

Olet jo ehkä huomannutkin, että CPC 464 pärjää erittäin hyvin näillä ominaisuuksillaan minkä hyvänsä samassa hintaluokassa olevan tietokoneen kanssa.

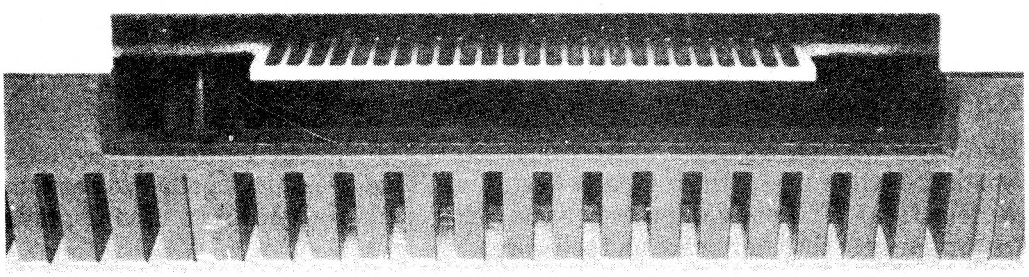
2. BASIC-TULKKI

Käytännöllisesti katsoen kaikissa kotitietokoneissa on BASIC-tulkki, jonka avulla käyttäjä voi tehdä omia ohjelmiaan pystyäkseen hyödyntämään laitteiston ominaisuuksia. Tietokoneen varusteisiin kuuluva "sisäänrakennettu" ohjelmointikieli (BASIC) on itsekin eräänlainen ohjelma. Se on erittäin monimutkainen ohjelma, jota on kehitetty yli miljoona miesvuotta siitä lähtien, kun BASIC "keksittiin" USA:ssa. BASIC (Beginners' All-purpose Symbolic Instruction Code) on varmasti maailman eniten käytetty tietokonekieli, ja siitäkin on lukuisia paikallisia "murteita" kuten kaikista muistakin kielistä.

CPC 464:n BASIC on yhteensopivuudeltaan eräs parhaista BASIC-kielen murteista, ja niinpä sillä voidaankin ajaa laaja valikoima yleisiä BASIC-ohjelmia, jotka on kirjoitettu käytettäviksi CP/M-käyttöjärjestelmän alaisuudessa. CPC 464:n BASIC on toiminnaltaan hyvin nopea, eli se suorittaa laskutoimitukset erittäin nopeasti. Vaikka sinua ei suuresti kiinnostaisikaan tietää, että jokin tietokone käyttää 0,05 sekuntia kertoakseen kolme viidellä ja esittääkseen tuloksen näytössä - verrattuna johonkin toiseen tietokoneeseen, jolla kuluu 0,075 sekuntia saman tehtävän suorittamiseen - niin on hyvä tietää, että ohjelma, joka piirtää grafiikkakuvioita kuvaruutuun, joutuu ehkä tekemään monta tuhatta yksinkertaista ja uudelleen toistuvaa laskutoimitusta. Tällöin 0,05 sekunnin ja 0,075 sekunnin välinen ero kasvaa merkittäväksi tekijäksi suorituskykyä arvioitaessa.

Tietokoneiden yhteydessä kuulee usein puhuttavan konekielisistä käskyistä ja ohjelmista. Konekielinen käsky on käskyn perusmuoto, joka voidaan ohjata suoraan prosessorille. Tällöin prosessorilta kuluu vähemmän aikaa ymmärtääkseen mitä pitää tehdä ja suorittaakseen vaaditun tehtävän. Tämä tapahtuu noin 5 - 15 kertaa nopeammin kuin jos vastaava tehtävä annetaan BASIC-kielisenä käskynä BASIC-tulkin välityksellä. Mutta toisaalta ohjelmointiaikaa voi kulua 5 - 50 kertaa enemmän konekielisen ohjelman laatimiseen verrattuna vastaavan BASIC-kielisen ohjelman tekemiseen.

CPC 464:ssä käytettävä AMSTRADin BASIC on eräs kotitietokoneiden nopeimmista ja laajimmista BASIC-kielistä. AMSTRADin BASICissa on koko joukko ominaisuuksia, joiden avulla kokenut BASIC-ohjelmoija pystyy välttämään BASIC-tulkin aiheuttamia hitaushaittoja ja muodostamaan hämmästyttävän dynaamisia näyttö- ja musiikkitehosteita.



3. Laajennettavuus

Useimmissa tietokoneissa on kiinnitetty huomiota niiden laajennettavuuteen, eli monet käyttäjät haluavat lisätä laitteistoon uusia laitteita. Tällaisia ovat esim. kirjoittimet, sauvaohjaimet, levyasemat, jne. Ihmeellistä kyllä, eräiden suosituimpien kotitietokoneidenkin kanssa on tarpeen käyttää ylimääräisiä liitäntäyksiköitä, ennen kuin edes yksinkertainen sauvaohjain tai kirjoitin voidaan liittää laitteistoon.

Tietokoneen ostaja ei aina osaa ajatella riittävän kauaskantoisesti voidakseen huomioida kaikki tulevat tarpeensa. Niinpä sellainen tietokone, jossa on valmiina kirjoittimen rinnakkaisliitäntä (Centronics) ja sauvaohjainliitäntä, voi tosiasiassa tulla pitkällä tähtäyksellä halvemmaksi.

CPC 464:n vakiovarusteisiin kuuluvat Centronics-tyyppinen kirjoitinliitäntä, kahden sauvaohjaimen liitäntämahdollisuus, stereoäänen lähtöliitäntä sekä monipuolinen laajennusväylä, johon voidaan liittää levyasemia, laajennus-ROM-piirejä, sarjaliitäntöjä (RS232), jne.

ROM-piiri (Read Only Memory = lukumuisti) on integroitu muistipiiri, joka sisältää pysyvästi tallennettuja ohjelmatietoja. CPC 464:n BASIC on tallennettu juuri tällaiselle ROM-piirille. On myös mahdollista toimittaa muita ohjelmia ROM-piireillä siten, että uudet ROM-piirit täydentävät koneen omaa ROM-piiriä tai korvaavat sen toiminnan kokonaan.

TV-peleissä käytettävät ohjelmat ovat yleensä moduulilla. Tällainen moduuli on itse asiassa ROM-piiri, joka on pakattu kestävään muovikoteloon ja varustettu sopivalla liittimellä, jotta moduuli voidaan laittaa helposti peliyksikköön ja irrottaa siitä yhä uudelleen ja uudelleen. Näin siis ROM toimii samalla tavalla kuin kasettikin toimittaen tietokoneelle ohjelmatietoja. Tiedot latautuvat kuitenkin ROM-piiriltä tietokoneen muistiin käytännöllisesti katsoen heti verrattuna useiden minuuttien latausaikaan, joka on varsin yleinen ladattessa pitkiä ohjelmia kasetilta. Täten ROM-piirin eräs tärkeimmistä eduista on juuri käyttömukavuus.

ROM-piiriä ei voida käyttää tietojen tallennukseen siten, että tietoja otettaisiin ROM-piiriltä ulos ja tallennettaisiin tai siirrettäisiin johonkin toiseen tietokoneeseen samalla tavalla kuin kasettinauhurin kanssa voidaan toimia.

Laajennettavuus varmistaa sen, että tietokoneesi pysyy ohjelmistoissa ja oheislaitteissa tapahtuvan kehityksen mukana jatkossakin. CPC 464-järjestelmä on laajennettavissa sangen monipuolisesti. Lisäksi laajennusmahdollisuudet on dokumentoitu kokonaan.

4. Äänet

Tietokoneen ääniominaisuudet määräävät sen, kuulostaako sen äänet hyttysen ininältä tyhjässä kaljatölkissä vai pystyykö se tuottamaan tyydyttävästi sähköisten soitinten ääniä.

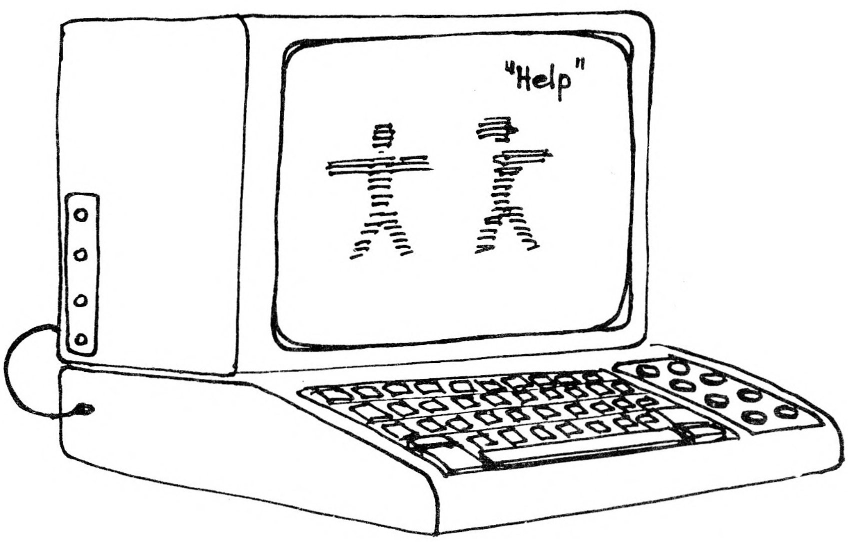
CPC 464:ssä on 3-kanavainen 8 oktaavin äänigeneraattori, jolla voidaan tuottaa varsin vaikuttavia musiikkitehosteita, koska sekä amplitudi että sävelvaippa (vastaa sävelkorkeutta) ovat täysin ohjelmoitavissa. Lisäksi ääni on jaettu stereofoniseksi eli yhdestä kanavasta saadaan vasemmanpuoleinen stereosignaali ja toisesta oikeanpuoleinen stereosignaali. Kolmas kanava on näiden kahden välillä oleva keskikanava. Tämä suo varsin mittavat mahdollisuudet tehdä ohjelmia, joissa äänitehosteet liittyvät saumattomasti yhteen koko kuvaruudun alueella esiintyviin kuvioihin arkadipelien tapaan.

Voit tietenkin itse viime kädessä päättää, mikä näistä CPC 464:n ominaisuuksista on sinulle kaikkein tärkein. Toivottavasti ne kaikki tuntuvat tärkeiltä, koska silloin saat tietokoneestasi parhaan hyödyn.

MITÄ TIETOKONEELLA VOI TEHDÄ JA MITÄ EI?

Monet ihmiset ihmettelevät usein, miksi edes CPC 464:n kaltainen uusinta tekniikkaa käyttävä edistyksellinen tietokone ei pysty tuottamaan tavallisen TV-kuvan kaltaisia kuvia.

Miksi tietokone ei pysty esimerkiksi jäljittelemään kuvaa, jossa ihminen kävelee kuvaruudun yli aivan luonnollisesti? Miksi kaikkien tietokoneiden kuvaruuduissa liikkuu "tikku-ukon" kaltaisia olioita?



Vastaus on sekä helppo että monimutkainen. Yksinkertainen vastaus on, ettei sinun kannata tuudittautua sellaiseen harhaluuloon, että tietokoneesi kuvaruudun erottelukyky olisi samanlainen kuin television kuvaruudussa. Television toiminta perustuu lineaarisen informaation käyttöön, jossa saadaan lähes rajaton erottelukykyalue kaikkien kirjon värien tummimmilla ja vaaleimmilla värisävyillä. Tämä tarkoittaa tietokonekielelle käännettynä sitä, että kokonaisen TV-kuvan tarvitsema "näyttömuistikapasiteetti" on noin 20 kertaa niin suuri kuin vastaavan kotitietokoneen näyttöruudullisen tarvitsema muistitila.

Eikä tässä vielä kaikki. Voidaksemme jäljitellä kuvan liikkeitä on välttämätöntä pystyä käsittelemään tätä valtavaa muistikapasiteettia suurella nopeudella (noin 50 kertaa sekunnissa). Tosin tämä on täysin mahdollista, mutta vain sellaisilla tietokoneilla, jotka maksavat muutaman tuhat kertaa enemmän kuin kotitietokoneet - ainakin tällä hetkellä!

Pienten tietokoneiden on pärjättävä suhteellisen pienillä näytön ohjaukseen käytettävillä muistikapasiteeteilla siihen asti, kunnes nopeiden muistipiirien hinta laskee ratkaisevasti (hintojen lasku on varsin ilmeistä). Tämä puolestaan aiheuttaa sen, että kuvan erottelukyky on pienempi ja liikkeet ovat karkeampia. Huolellinen laitteiden suunnittelu ja laadukkaat ohjelmat helpottavat varsin pitkälle tilanteen aiheuttamia hankaluuksia, mutta edessä on vielä melko pitkä matka kuljettavaksi, ennen kuin huokeat tietokoneet pystyvät tuottamaan sulavia liikkeitä ja elävämpiä kuvia ainakin samalla tavalla kuin mihin piirretyt elokuvat pystyvät nykyään.

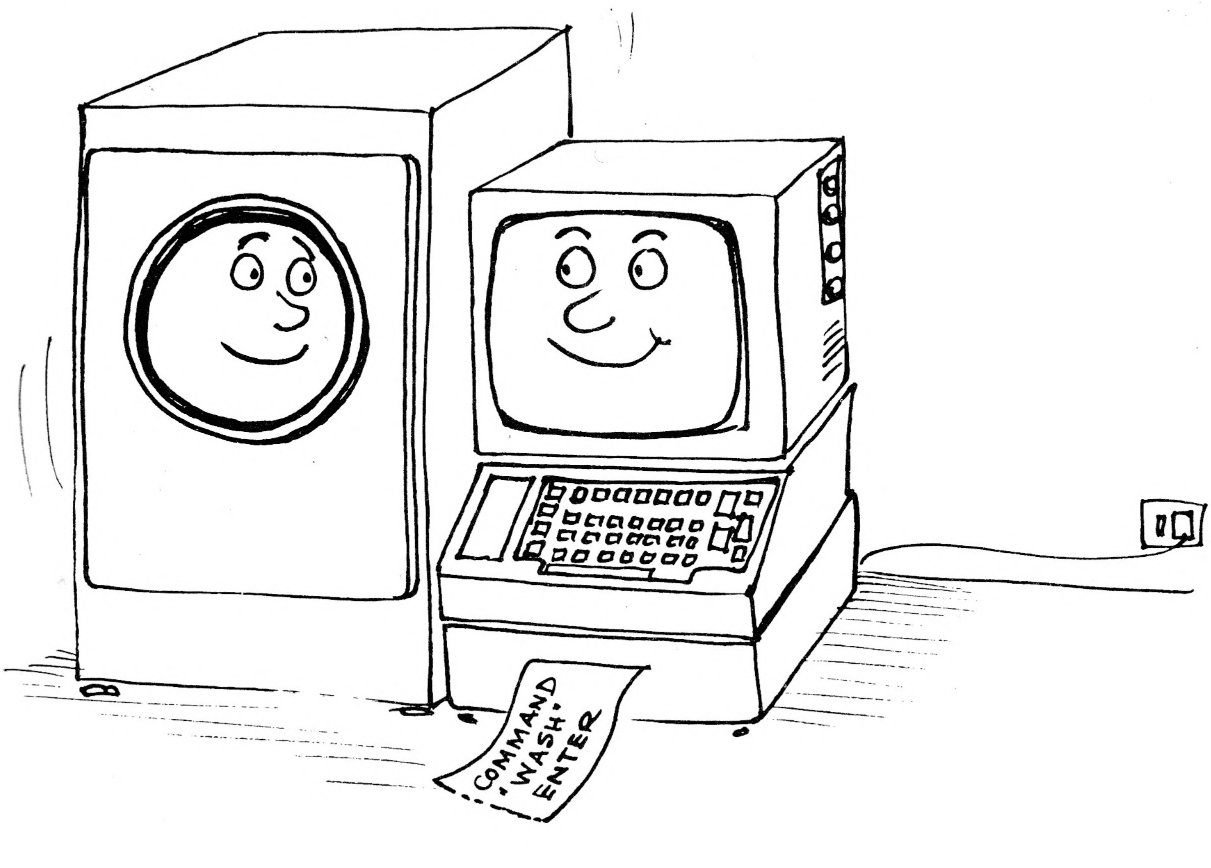
Toinen varsin yleinen ihmettelyn aihe on, miksi tietokoneeseen ei voida muitta mutkitta näppäillä tavallista tekstisivua?

Vaikka tietokone näyttääkin näyttöpäätteellä varustetulta kirjoituskoneelta, ei tämän seikan pidä antaa hämätä itseään. Tietokoneen kuvaruutu ei ole pala sähköistä paperia, vaan se on viestintäväline, jonka avulla käyttäjä voi "keskustella" koneen muistissa olevan ohjelmointikielen (ja ohjelmien) kanssa.

Tietokone yrittää itsepintaisesti tulkita kaikki näppäimistöltä näppäilemäsi merkit ohjelmakäskyiksi, ellet nimenomaan ilmoita tietokoneelle, että tarkoitat jotain muuta. ENTER-näppäintä painaessasi tietokone katsoo mitä olet näppäillyt, ja jos siinä ei ole mitään järkeä koneen BASICin mielestä, se hylkää näppäilemäsi merkit ja tulostaa kuvaruutuun virheilmoituksen **Syntax Error** .

On tietenkin mahdollista, että tietokoneesi muistissa parhaillaan oleva ohjelma sattuu olemaan tekstinkäsittelyohjelma, jolloin voit tietenkin näppäillä minkälaisia sanoja hyvänsä ja painaa ENTER-näppäintä. Tällaisessa tapauksessa voit luonnollisesti jatkaa näppäilyä ikään kuin järjestelmä toimisi elektronisen kirjoituskoneen sähköisenä "paperina". Sinun on kuitenkin ensin ladattava tekstinkäsittelyohjelma tietokoneen muistiin kasettinauhurilla ennen kuin voit tehdä mitään tällaista.

Tietokoneessa näyttää tavallaan yhdistyvän useampia koti- ja toimistoympäristössä tutuksi tulleita laitteita, kuten TV:n kaltainen kuvaruutu, näppäimistö ja kasettinauhuri. Tästä huolimatta sinun täytyy muistaa, että yhtäläisyydet ovat yleensä täysin näennäisiä, ja että tietokone on aivan oma persoonallisuutensa tietyistä ulkoisista yhtäläisyyksistään huolimatta.



TIETOKONESANASTOA

Sanastoon on kerätty eräitä yleisimpiä tietokoneiden yhteydessä esiintyviä sanoja. Kaikki sanat eivät kuitenkaan ole vakiintuneet edes englanninkielisissä muodoissaan, joten käännöstermejä esiintyy käytössä sitäkin enemmän. On muistettava, ettei mitään ehdotonta ja ainoaa oikeaa käännöstä ole vaan tärkeintä on se, että asia tulee ymmärretyksi. Sanastossa on pyritty käyttämään yleisimpiä käyttöön vakiintuneita termejä.

Aakkosnumeerinen ks. Alphanumeric

Accumulator akku

Akku on tietokoneen keskusyksikkönä toimivassa mikroprosessorissa oleva muistipaikka, johon voidaan tallentaa tietoja väliaikaisesti käsittelyn aikana. Akkua käytetään laajalti konekieliohjelmoinnissa, mutta BASIC-käyttäjien ei tarvitse vaivata päätään sen olemassaololla.

Acoustic coupler akustinen modeemi
Akustinen modeemi on elektroninen lisälaite, jolla tietokone voidaan kytkeä puhelimen kuulokkeeseen. Näin tietokone pystyy olemaan yhteydessä tavallisen puhelinverkon välityksellä esim. muihin tietokoneisiin, tietopankkeihin yms. tietopalveluihin. Tällä tavalla toisiinsa kytketyt tietokoneet voivat vaihtaa keskenään tietoja ja ohjelmia.

Address osoite
Osoite on käskyyn kuuluva luku, jolla määritellään tietokoneen muistin yksittäinen muistipaikka. Osoitteen avulla voidaan aina valita haluttu muistipaikka, jolloin sen sisältö voidaan etsiä ja lukea. RAM-muistin tapauksessa voidaan muistipaikassa oleva tieto lukea ja tallentaa uudelleen siihen tehtyjen muutosten jälkeen. ROM-muistin tietoja voidaan vain lukea.

Adventure game seikkailupeli
Toiset pitävät seikkailupeleistä ja toiset taas eivät. Seikkailupelit ovat tekstipohjaisia tietokonepelejä, joissa pelaajalle esitetään kuvaruudussa erilaisia tehtäviä ja annetaan tilanteisiin kuuluvia tietoja, joita sitten yritetään käyttää hyväkseen kussakin tilanteessa. Pelin tavoitteena on yrittää päästä esim. labyrintin läpi suoriutuen kaikista sen varrella esitetyistä tehtävistä.

Algorithm algoritmi
Algoritmi on vain jonkin monimutkaisen kaavan tai summan hienolta kalskahtava nimi. Kysymys on yleensä melko suuresta joukosta perättäisiä loogisia ja aritmeettisia vaiheita päämääränä tietyn laskentatehtävän suorittaminen.

Aliohjelma ks. Subroutine

Alkeisharjoitus ks. Primer

Alkulataus ks. Booting or Bootstrapping

Alphanumeric aakkosnumeerinen
Aakkosnumeerisia merkkejä ovat kirjain- ja numeromerkit. Grafiikkamerkit eivät siis kuulu aakkosnumeerisiin merkkeihin.

Akku ks. Accumulator

Akustinen modeemi ks. Acoustic coupler

ALU aritmeettislooginen yksikkö
Aritmeettislooginen yksikkö on osa mikroprosessoria, ja sen tehtävänä on suorittaa aritmeettisia ja loogisia operaatioita. Myöskään ALU:n toiminnasta ei tarvitse välittää muuta kuin konekielisessä ohjelmoinnissa.

Alustaa ks. Initialise

AMSOFT
AMSOFT on AMSTRADin tukiosasto, jonka erikoisalaan kuuluu tietokoneiden ohjelmistojen, oheislaitteiden ja julkaisujen toimittaminen. AMSOFTin palvelut on tarkoitettu erityisesti CPC 464:n käyttäjien tueksi.

A/D
Analogue analoginen
Esim. analoginen signaali on sellaista, jossa jännitteen muutokset alku- ja loppuarvojen välillä tapahtuvat yhtäjaksoisena ilmiönä eikä portaittain. Tietokoneiden toiminta perustuu puolestaan digitaalitekniikkaan, kun taas useimmat elävässä elämässä tapahtuvat ilmiöt ovat luonteeltaan analogisia. Niinpä tietokoneen on suoritettava A/D-muunnos (analogia/digitaali-muunnos), ennen kuin se pystyy käsittelemään analogisesta lähteestä saatavia tietoja.

Animation animaatio
Piirretyt elokuvat ovat ehkä parhaiten tunnettu animaation muoto. Tietokoneanimaatio perustuu liikkuvien grafiikkakuvioiden käyttöön, joilla halutaan jäljitellä erilaisia olioita ja liikkeitä.

Applications program sovellus- tai sovellutusohjelma
Ohjelma, joka on laadittu (sovelluttu) johonkin tiettyyn erikoistehtävään sen sijaan, että ohjelmaa voitaisiin käyttää monessa eri tehtävässä yleiskäyttöisten ohjelmien tapaan. Tällaisia erikoisohjelmia ovat esim. assemblerohjelmat, kirjoittimien ohjausohjelmat, budjetointiohjelmat, kirjanpitoohjelmat, jne.

Apuohjelma ks. Utility program

Arcade game arkadipeli
Arkadipeleiksi kutsutaan sellaisia tietokone- ja videopelejä, joissa esim. yritetään puolustautua avaruusolioiden hyökkäystä vastaan, tai väsymättömät hirviöt ajavat takaa pitkin sokkeloita joista yritetään poimia mukaan kaikenlaisia aarteita. Arkadipeleissä pelaaja siis yrittää ohjata pelin kulkua kaikenlaisia "kuolemia" vältellen. Arkadipelit ovat yleisesti ottaen hauskoja ja monasti hyviä refleksien kannalta, mutta niistä on varsin vähän hyötyä tietokonetekniikasta syvällisemmin kiinnostuneille.

Architecture arkkitehtuuri
Arkkitehtuuriksi kutsutaan sitä tapaa, jolla tietokoneen dataväylät, oheislaitteet ja CPU (keskusyksikkö) on sijoitettu toisiinsa nähden toiminnallisessa mielessä. Tämän aiheen ei pitäisi liikuttaa kovinkaan paljoa sanaston lukijaa ainakaa vielä tässä vaiheessa.

Argument argumentti
Itsenäinen muuttuja. Esim. lausekkeessa x + y = z ovat x ja y argumentteja.

Aritmeettislooginen yksikkö ks. ALU

Arkadipeli ks. Arcade Game

Arkkitehtuuri ks. Architecture

Array taulukko (t. matriisi)
Taulukko on yleensä 2-ulotteinen matriisi (ruudukko), johon tietoja tallennetaan käyttämällä osoitteena vaakasuunnan ja pystysuunnan koordinaattiarvoja. Tietokoneessa taulukot voivat olla myös useampiulotteisia, kuten tässä käsikirjassa on toisaalla selostettu.

Artificial intelligence (AI) tekoäly
Tekoälyksi kutsutaan sellaista ohjelmointitekniikkaa, jossa ohjelma pystyy oppimaan aikaisemmista kokemuksistaan.

ASCII
ASCII on lyhenne sanoista American Standard Code for Information Interchange. Se on yleisesti käytetty tapa esittää numeroita, kirjaimia ja muita symboleja, jotka voidaan näppäillä tietokoneen näppäimistöltä tai saada käyttöön tietyillä käskyillä. CPC 464:n ASCII-koodit on lueteltu liitteessä III.

Assembler assembler-ohjelma (t. symbolisen konekielen käännösohjelma)
Kääntää symbolisen konekielen konekielelle (bittimuotoon).

Assembly language symbolinen konekieli
Tätä kieltä käytetään konekielisten ohjelmien tekemiseen. Siinä konekieliset käskyt saadaan käyttöön muistikkaiden (mnemonics) avulla. Muistikkaat ovat puolestaan kirjaimia tai kirjainyhdistelmiä, jotka ovat yleensä lyhenteitä käskystä tai toiminnosta, jonka vastaavan konekielisen aliohjelman halutaan suoritettavan.

Avainsana ks. Keyword

Bar code viivakoodi
Viivakoodeja käytetään jo varsin yleisesti esim. elintarvikepakkauksissa. Viivakoodit on tehty tietokoneella luettavaan muotoon, jolloin esim. elintarvikkeista saadaan selville niiden hinta- ja tuotetiedot. Viivakoodin lukemiseen käytetään yleisesti kynän muotoista optista laitetta, jota kutsutaan viivakoodin lukukynäksi (wand). Lukukynän optiset toiminnot voivat perustua esim. pienitehoisen laserin käyttöön.

Base kantaluku
Kantaluku on yksi matematiikan peruskäsitteistä. Kaikkinainen numeromuodossa tapahtuva esitys perustuu johonkin kantalukuun. Binäärijärjestelmän kantaluku on 2, desimaalijärjestelmän kantaluku on 10 ja heksadesimaalijärjestelmän kantaluku on 16. Aihetta käsitellään tarkemmin liitteessä II.

BASIC
BASIC on lyhenne sanoista Beginners' All-purpose Symbolic Instruction Code, ja se on luonteeltaan tulkittava ohjelmointikieli. BASIC suunniteltiin alunperin mahdollisimman helpoksi oppia ja käyttää, ja niinpä sitä käytetäänkin lähes kaikissa kotitietokoneissa. BASICille on luonteenomaista, että ohjelmia voidaan kokeilla missä vaiheessa hyvänsä ja tehdä niihin tarvittavia muutoksia ja lisäyksiä. Tietokoneen BASIC-tulkki tulkitsee käskyn kerrallaan eli muuttaa kunkin BASIC-käskyn mikroprosessorin ymmärtämään muotoon, jotta halutut toiminnot saataisiin aikaan. BASIC-ohjelmat eroavat kääntäjätyyppisistä ohjelmista (compile), joissa koko ohjelma on ensin ajettava, ennen kuin ohjelman toimivuutta päästään tarkemmin arvioimaan.

Baud baudi
Baudia käytetään nopeuden mittayksikkönä siirrettäessä digitaalitietoja sarjamuotoisissa tietoliikennejärjestelmissä. 1 baudi = 1 bitti sekunnissa.

BCD
BCD on lyhenne sanoista Binary Coded Decimal ja se tarkoittaa binäärikoodattua desimaalilukua. Se on siis desimaalilukujen koodausjärjestelmä, jossa kutakin numeroa vastaa neljän binäärinumeron (bitin) ryhmä.

Benchmark program (testiohjelma)
Tämä on tietyntyyppinen testiohjelma, jota käytetään vertailtaessa eri tietokoneiden nopeutta, tehokkuutta ja tarkkuutta. Esimerkki neliöjuuri luvusta 99 999 korotettuna toiseen potenssiin.

Binary binäärimuoto
(Katso kohtaa Base; kantaluku). Binäärijärjestelmän lukujen kantaluku on 2, jossa kaikki luvut muodostetaan käyttäen kahta binäärinumeroa 0 ja 1.

Binary number binääriluku
Binääriluku on binäärimuodossa esitetty luku. CPC 464:n ohjelmoinnissa binäärilukujen merkitsemiseen käytetään etuliitettä &X. Esimerkki &X0101, joka on desimaalimuodossa 5.

Bit bitti
Bit on lyhenne sanoista BInary digiT eli binäärinumero. Bitti voi saada vain jomman kumman kahdesta arvosta, eli se voi olla joko 0 tai 1. Kaikki binäärijärjestelmä luvut koostuvat juuri biteistä. Aihetta käsitellään myös liitteessä II.

Bit significance bitin merkitsevyys

Yksi tavu koostuu kahdeksasta bitistä. Joskus tieto otetaan ulos jostakin luvusta tutkimalla sen binäärimuodon kunkin kahdeksan bitin tilaa. Tällöin vain näiden bittien tila merkitsee. Voi olla myös niin, että vain esim. neljännen bitin tila on merkitsevä, eikä muiden bittien tilasta välitetä. Luvun vastaava desimaalimuoto saattaa olla täysin merkityksetön.

Boolean algebra Boolen albegra
Boolen algebraa käytetään loogisissa vertailulausekkeissa, joiden tulokseksi voidaan saada vain kaksi vastausta: tosi tai epätosi. Niitä vastaa yleensä bitit 0 ja 1.

Booting or Bootstrapping alkulataus
Ohjelmat ja käyttöjärjestelmät eivät lataudu itsestään, vaan ne ladataan pienellä alkulatausohjelmalla, joka on yleensä tallennettu ROM-piirille. Alkulatausohjelma käynnistää latausprosessin tietystä muistipaikasta.

Buffer puskuri
Puskuriksi kutsutaan väliaikaista tallennusaluetta, jossa säilytetään järjestelmän toisesta osasta toiseen osaan siirrettäviä tietoja. Esimerkkinä tiedonsiirto kasetin ohjausohjelmistolta tietokoneen keskusyksikölle (CPU:lle) ja pää-RAM-piirille. Puskuri säätelee tapaa, jolla tietoja ohjataan eri nopeuksilla toimivien laitteiden välillä. Esimerkkinä eri nopeuksilla toimivista laitteista ovat esim. modeemi ja kirjoitin.

Bug ohjelmavirhe
Ohjelmavirheiksi kutsutaan monenlaisia ohjelmissa esiintyviä ongelmia. Ongelma voi aiheutua esim. siitä, että ohjelmaa käytetään odotuksista poikkeavalla tavalla (jos esim. painat neljää näppäintä kerralla, voi kuvaruudun väri vaihtua). Suuremman luokan ongelmia aiheuttavat sellaiset virheet, jotka peruuttamattomasti turmelevat ohjelman ja pyyhkivät kaikki tiedot muistista.

Bus väylä
Väyläksi kutsutaan tiettyä linjaryhmää, joka voi olla tietokoneen sisällä tai yhdistää tietokoneen ulkomaailmaan. Väylä välittää tietoja CPU:n tilasta, RAM-muistin sisällöstä ja muista laitteiston ominaisuuksista. CPC 464:n väylä on kytketty suurempaan kahdesta tietokoneen kotelon takaosassa olevasta korttiliittimestä. CPC 464 kytketään ulkomaailmaan juuri tämän väylän kautta.

Byte tavu
Tavu koostuu kahdeksasta bitistä, ja 8-bittisen CPU:n pienin muistialkio - johon voidaan tallentaa tietoa ja lukea sitä sieltä - on juuri yhden tavun kokoinen. Asiaa käsitellään tarkemmin liitteessä II.

CAD tietokoneavusteinen suunnittelu
CAD on lyhenne sanoista Computer Aided Design. Yleensä sillä tarkoitetaan sovellutusta, jossa suunnittelija käyttää apunaan tietokoneen laskentatehoa ja graafisia ominaisuuksia, eli käyttää tietokonetta eräänlaisena elektronisena piirustuspöytänä, vaikkakin suunnittelutyöhön sisältyy myös usein paljon tietokoneella suoritettavia laskutoimituksia.

CAE tietokoneavusteinen opetus
CAE on lyhenne sanoista Computer Aided Education. Tämä tarkoittaa sananmukaisesti tietokoneen käyttämistä apuna opetuksessa.

Cartridge moduuli (t. kasetti)
Kasetilla tarkoitetaan tietokoneiden yhteydessä yleensä C-kasettia. Tässä kuitenkin on kysymys ohjelmamoduulista, joka on yleensä muovikoteloon pakattu muistipiiri (ROM), joka sisältää ohjelman tai ohjelmia. Ohjelmamoduulit ovat usein tietokonemerkkikohtaisia, jolloin tietokoneessa on liitäntä, johon ohjelmamoduuli työnnetään. Ohjelmamoduulilla toimitettavat ohjelmat latautuvat ja toimivat yleensä huomattavasti nopeammin ja helpommin kuin vastaavat C-kaseteilla toimitettavat ohjelmat, mutta haittapuolena on niiden huomattavasti korkeampi hinta.

Cassette kasetti
Kaseteilla tarkoitetaan yleensä C-kasettia, joille on tallennettu tietokoneohjelmia. Kasetti on varsin huokea mutta hidas tallennusväline.

Character merkki
Merkeiksi sanotaan kaikkia symboleja, jotka tietokoneella voidaan muodostaa ja tulostaa esim. näyttöön. Merkkejä ovat kirjaimet, numerot ja grafiikkasymbolit (katso liitettä III).

Character cell merkkisolu
Merkkisoluksi kutsutaan yhden merkin muodostamiseen tarvittavaa pistematriisia (8 x 8 pistettä), ja merkki saadaan näkyviin kuvaruudussa sytyttämällä merkkisolun tietyt pisteet (katso liitettä III).

Character set merkkivalikoima
Merkkivalikoimalla tarkoitetaan kaikkia tietokoneen tai kirjoittimen merkkejä eli kirjaimia, numeroita ja grafiikkasymboleja. Tietyn merkin esiintyminen tietokoneessa ei välttämättä merkitse sitä, että se saadaan tulostetuksi millään kirjoittimella.

Character string merkkijono
Merkkijono on joukko perättäisiä merkkejä. Merkkijono voidaan tallentaa ja sitä voidaan käsitellä ohjelmassa yhtenä kokonaisuutena. Esimerkkinä merkkijonosta mainittakoon vaikkapa sana tai joukko sanoja. Merkkijonoa voidaan käyttää myös muuttujana (määritellään dollarinmerkillä **$**), jolloin puhutaan merkkijonomuuttujasta. (Ks. String variable).

Chip siru
Kaikenlaisia monoliittisia elektronisia integroituja piirejä kutsutaan varsin yleisesti - joskin virheellisesti - siruiksi. Siru on kuitenkin tosiasiassa pieni erikoistekniikalla valmistettu piipala, jolle varsinainen integroitu piiri valmistetaan.

Clock kello
Kello on tietokoneen pääajastin, josta saatavilla signaaleilla tietokoneen eri toiminnot tahdistetaan ja ajastetaan. Reaaliaikaisesta kellosta (jollainen CPC 464:ssä on) saadaan tunnit, minuutit, sekunnit, jne.

Code koodi
Koodi-sanalla on sangen monta merkitystä, mutta useimmiten sillä ohjelmoinnin yhteydessä tarkoitetaan konekoodia eli konekieltä (machine code tai machine language).

Command komento (t. käsky)
Komennolla tarkoitetaan yleensä ohjelmointikäskyä.

Compiler kääntäjäohjelma
Kääntäjäohjelma on sangen monimutkainen ohjelma, joka muuttaa kokonaisia esim. BASICin kaltaisilla tulkittavilla korkeantason lausekielillä kirjoitettuja ohjelmia mikroprosessorin ymmärtämälle konekielelle. Kokonaan konekielelle käännetty ohjelma toimii merkittävästi nopeammin kuin käskyittäin BASIC-tulkilla tulkittava BASIC-ohjelma.

Computer generations tietokonesukupolvet
Tietokonetekniikan kehityksessä on erotettavissa tiettyjä kehitysvaiheita, ja kuhunkin näihin kehitysvaiheisiin kuuluvia tietokoneita kutsutaan kyseisen sukupolven tietokoneiksi.

Computer literacy uusi lukutaito
Uudesta lukutaidosta puhuttaessa tarkoitetaan tietokoneiden ymmärtämistä.

CP/M
CP/M on Digital Research-nimisen yrityksen luoma standardi, ja se koskee levynkäyttöjärjestelmiä. Käytännössä tämä tarkoittaa sitä, että lähes kaikki CP/M-ohjelmat toimivat niissä tietokoneissa, joissa on CP/M-käyttöjärjestelmä. CPC 464:ssä on CP/M 2.2-käyttöjärjestelmä käytettäessä levyasemaa.

CPU keskusyksikkö
CPU on lyhenne sanoista Central Processing Unit. Keskusyksikkö on tietokonejärjestelmän keskeisin osa, ja sen tehtävänä on tulkita tietokoneelle annettavat käskyt ja huolehtia siitä, että annettuja käskyjä noudatetaan. Mikrotietokoneessa CPU on itse mikroprosessori.

Cursor kohdistin
Kohdistimeksi nimitetään sitä näytössä näkyvää merkkiä joka osoittaa, mihin kohtaan seuraava merkki kirjoittuu kuvaruutuun.

Cursor control keys kohdistimen ohjausnäppäimet
Nämä näppäimet ovat kohdistimen liikutteluun käytettäviä näppäimiä. Näppäimillä voidaan ohjata kohdistinta eri puolille kuvaruutua, ja niitä käytetään sangen usein erilaisten pelien suuntanäppäiminä. Näppäimet on merkitty nuolimerkinnöin, ja ne sijaitsevat CPC 464:n erillisen numeronäppäimistön yläpuolella.

Daisy-wheel printer kirjasinkiekkokirjoitin
Kirjasinkiekkokirjoitinta kutsutaan laatukirjoittimeksi, koska sillä saadaan aikaan "kirjelaatuista" tulostusjälkeä. Merkki saadaan tulostumaan paperille siten, että kirjasinkiekossa olevaa kirjainmerkkiä lyödään pienellä "vasaralla", ja isku välittyy värinauhan läpi paperille.

Database tietokanta
Tietokannaksi kutsutaan taulukkomuodossa tallennettua tietoa, joka voi olla minkätyyppistä tietoa hyvänsä, kunhan se vain on tietokoneen luettavassa muodossa.

Data capture tietojen keräys
Tietojen keräys tarkoittaa sitä, että tietoja kerätään erilaisista keskustietokoneeseen jollain tavalla kytketyistä ulkopuolisista tietolähteistä.

Debugging virheiden poisto
Virheiden poisto tarkoittaa sitä, että ohjelmassa ongelmia aiheuttavat virheet yritetään löytää ja korjata.

Decimal notation desimaalimuoto
Desimaalimuoto tarkoittaa esitystapaa, jossa luvut esitetään käyttäen numeroita 0 - 9, jotka edustavat ykkösiä, kymmeninä, satoja, tuhansia, jne. Lukujen kantaluku on 10.

Diagnostic diagnostiikka
Diagnostiikka on tietokoneen automaattisesti suorittama toiminto, jolla tietokone tunnistaa ja yksilöi ohjelmasta löytämänsä virheen ja ilmoittaa siitä käyttäjälle virheilmoituksella.

Digital digitaalinen
Esimerkiksi digitaalinen signaali on sellaista, jonka jännite muuttuu tarkasti määriteltyjen portaiden verran eli portaittain eikä jatkuvasti, kuten analogisessa signaalissa. Digitaalinen on siis analogisen vastakohta.

Digitiser digitoija (t. digitointilaite)
Digitoijaksi kutsutaan yleensä laitetta, jolla analogisessa muodossa olevaa tietoa voidaan syöttää tietokoneelle. Digitoijista puhutaan yleensä graafisten tablettien (digitointipöytien) yhteydessä.

Disk levy (t. tietolevy)
Tietolevyksi (tai levykkeeksi) kutsutaan tietojen tallennusvälineenä käytettävää ohutta pyöreää muovilevyä, joka on pinnoitettu toiselta tai molemmilta puoliltaan magneettisella oksidikerroksella. Tietolevyä säilytetään aina omassa pahvista tai muovista valmistetussa suojakotelossaan, ja levyaseman luku/kirjoituspää pääsee kosketukseen magneettipinnan kanssa kotelossa olevan lukuaukon kautta. Katso myös kohtia Floppy disk (levyke) ja Winchester (kiinteä levy).

Disk drive levyasema
Levyasemaksi kutsutaan laitetta, joka tallentaa tietoja levyasemassa pyörivän tietolevyn magnetisoidulle pinnalle ja lukee (palauttaa takaisin käyttöön) levylle tallennetut tiedot.

Documentation dokumentointi
Dokumentointi tarkoittaa tietokoneiden tai ohjelmien mukana toimitettuja käsikirjoja, joissa selitetään tietokoneiden tai ohjelmien toimintaa ja käyttöä.

Download vastaanottaa lataustietoja
Kun tietoja siirretään tietokoneelta toiselle, nimitetään tietoja vastaanottavaa konetta yleensä vastaanottavaksi koneeksi (downloading). Toisessa päässä olevaa konetta kutsutaan lähettäväksi koneeksi (uploading).

DOS
DOS on lyhenne sanoista Disk Operating System = levynkäyttöjärjestelmä. Se on ohjelmisto, joka ohjaa levyaseman kaikkia toimintoja.

Dot matrix pistematriisi
Pistematriisi on suorakaiteen muotoinen pisteruudukko, jonka avulla merkki saadaan muodostetuksi valitsemalla tietyt matriisin pisteet.

Dumb terminal tyhmä pääte
Tyhmä pääte on sellainen tietokonepääte, joka toimii pelkästään syöttö/tulostusvälineenä käsittelemättä millään tavalla sen kautta kulkevia tietoja. Eräs tyhmä pääte on ns. älytön pääte (mindless terminal) vastakohtana älykkäälle päätteelle. Siinä ei ole edes näytön ohjauselektroniikkaa, ja kuvaruudun näyttötiedot tuodaan siihen puhtaana videosignaalina.

Editing korjailu (t. editointi)
Korjailu tarkoittaa tietojen, ohjelman tai tekstin korjailua tai muuttamista.

Editor toimitusohjelma (t. editori)
Tämä on ohjelma, joka on tavallisesti tietokoneen ROM-piirillä. Ohjelman avulla voidaan toimittaa eli luoda, korjailla ja muutella tietoja, ohjelmia tai tekstejä.

Erillinen ks. Off line

Erotin ks. Separator

Erottelukyky ks. Resolution

EPROM
EPROM on lyhenne sanoista Erasable Programmable Read Only Memory = pyyhittävä ja ohjelmoitava lukumuisti. EPROM on muuten samanlainen kuin PROM-piirikin, mutta EPROM-piirille tallennetut tiedot voidaan pyyhkiä ultraviolettivalolla, jonka jälkeen muistipiiriin voidaan ohjelmoida uudet tiedot. EEPROM on muuten samanlainen, mutta se voidaan pyyhkiä sähköisesti.

Etumerkitön luku ks. Unsigned number

Expression lauseke
Lauseke on yksinkertainen tai monimutkainen kaava, jota käytetään ohjelmassa tiedoilla suoritettavaan laskutoimitukseen. Lauseke määrittelee yleensä datatyypin, jota se pystyy käsittelemään.

Fifth generation computers viidennen sukupolven tietokoneet
Näillä tietokoneilla tarkoitetaan yleensä suuria keskustietokoneita, joiden on arveltu pystyvän ohjelmoimaan itse itseään käyttäen hyväksi tekoälyssä tapahtunutta kehitystä.

File tiedosto
Tiedostoksi kutsutaan tiettyä nimellä määriteltyä tietojoukkoa, joka on yleensä tallennettu kasetille tai tietolevylle. Joissakin tietokoneissa on myös RAM-tiedostoja.

Firmware kiinteä laiteohjelmisto
Kiinteäksi laiteohjelmistoksi kutsutaan ROM-muistissa olevaa ohjelmistoa eli puhtaan ohjelmiston ja puhtaan laitteiston risteytystä.

Fixed-point number kiintoluku
Tämä on luku, jossa desimaalipiste on aina tietyssä kiinteässä kohdassa lukua esitettäessä, käsiteltäessä ja tallennettaessa.

Floating-point number liukuluku
Liukuluku on reaaliluku, jonka desimaalipisteen paikkaa tietokone voi tarvitessaan muuttaa käsitellessään ja tallentaessaan lukuja. Menetelmä on erityisen käyttökelpoinen käsiteltäessä suuria lukuja.

Floppy disk levyke (t. tietolevy)
Tietolevy on irrallinen 5,25 tai 8 tuuman läpimittainen magneettilevy, jota käytetään levyasemassa tietojen tallennukseen. Tietolevy on neliskulmaisen suojakotelon sisällä. Tietolevyn tallennuskapasiteetti on huomattavasti suurempi kuin kasetilla, ja se on paljon nopeampi mutta kalliimpi.

Flowchart vuokaavio (t. kulkukaavio)
Vuokaavioksi kutsutaan ohjelmavaiheiden ja loogisten tapahtumien kulkua kuvaavaa kaaviomaista esitystä. Vuokaaviosta voidaan seurata tapahtumien järjestystä ohjelman suorituksen aikana.

Forth
Forth on nopea ohjelmointikieli, joka nopeudeltaan ja monimutkaisuudeltaan sijoittuu lausekielen ja konekielen välille. Forth ei ole tarkoitettu aloittelijoille.

Function key toimintonäppäin
Toimintonäppäimeksi kutsutaan tietokoneen näppäimistön näppäintä, jolle on nimetty jokin tietty tehtävä tai toiminto, joka voi korvata näppäimen alkuperäisen toiminnon tai se voi olla kyseisellä näppäimellä saatava lisätoiminto. CPC 464:ssä on koko joukko näppäimiä, jotka voidaan määritellä toimintonäppäimiksi. Täten yhdellä näppäimen painalluksella saadaan käyttöön jopa 32 merkkiä käsittävä merkkijono, joka on yleensä joukko perättäisiä (kaksoispisteellä erotettuja) käskyjä tai modeemien ja kirjoittimien tapaisten oheislaitteiden ohjaamiseen käytettäviä käskyjä.

Gate veräjä
Loogiset veräjät päästävät tietoa lävitseen tiettyjen ehtojen täyttyessä. Veräjiä on montaa eri tyyppiä (AND = JA, OR = TAI, XOR = POISSULKEVA TAI, jne.). Aihe liittyy Boolen algebraan.

Graphics grafiikka
Grafiikaksi kutsutaan sitä tietokoneen näytön osaa, joka ei liity merkkien tulostukseen. Grafiikkaa ovat siis esim. viivojen, ympyröiden, kaavioiden, jne. piirtäminen. Grafiikkakuvioista on mahdollista saada myös paperikopio, kunhan tietokoneeseen on vain liitetty sopiva kirjoitin.

Graphics character grafiikkamerkki
Grafiikkamerkiksi kutsutaan erityistä symbolia, jota voidaan käyttää apuna erilaisten kuvien ja hahmojen luonnissa. CPC 464:ssä on kattava valikoima grafiikkamerkkejä, ja niitä kuvataan liitteessä III.

Graphics cursor grafiikkakohdistin
Grafiikkakohdistin on samanlainen kuin tekstikohdistinkin, mutta grafiikkakohdistinta käytetään grafiikkaruudun osoittamiseen. CPC 464:ssä grafiikkakohdistin on näkymätön mutta silti korvaamaton apuväline, jolla grafiikkakuviot saadaan tulostetuiksi juuri oikeisiin kuvaruudun kohtiin. Tähän ei pidä sekoittaa grafiikkamerkkejä (liite III), jotka kuuluvat osana merkkivalikoimaan ja tulostuvat tekstikohdistimen paikkaan.

Graphics mode grafiikkatila
Vanhemmat tietokoneet piti aikaisemmin erikseen asettaa joko merkkien käsittelytilaan tai grafiikkatilaan. Nykyiset mikrotietokoneet pystyvät vallan mainiosti käsittelemään tekstiä ja grafiikkaa samanaikaisesti.

Graphics tablet grafiikkatabletti (t. piirtopöytä)
Laite, jolla tietyn kuvan tai piirustuksen koordinaattipisteet saadaan siirretyksi tietokoneelle käsittelyä varten. Liittyy myös aiheeseen A/D.

Handshaking kättely
Kättelysignaaleiksi kutsutaan sarjaa perättäisiä sähköisiä signaaleja, jotka käynnistävät, tarkistavat ja tahdistavat kahden tietokoneen tai tietokoneen ja oheislaitteen välisen tiedonsiirron.

Hard copy paperikopio (t. paperituloste)
Paperikopioksi kutsutaan ohjelman, tekstin tai grafiikkanäytön paperitulostetta. Hard copy ei siis häviä vaikka virta katkaistaan tietokoneesta, mutta kuvaruudussa näkyvä vastaava "Soft Copy" häviää, kun virta katkaistaan.

Hardware laitteisto
Laitteistoksi kutsutaan tietokonejärjestelmän kaikkia elektronisia ja mekaanisia osia eli kaikkea muuta, mikä ei ole ohjelmistoa (software) tai kiinteää laiteohjelmistoa (firmware).

Hexadecimal notation heksadesimaalimuoto
Heksadesimaalijärjestelmään kuuluvien lukujen kantaluku on 16. Heksadesimaalilukuja merkitään CPC 464:ssä etuliitteellä & tai &H. Aihetta käsitellään tarkemmin liitteessä II.

High-level language lausekieli
Lausekieliksi kutsutaan kieliä, jotka on kirjoitettu lausemuotoon, jossa eri tehtäviä kuvataan erilaisilla lauseilla. Lausekieli on hitaampaa kuin koneenläheiset ohjelmointikielet (esim. konekieli), mutta lausekieltä on paljon helpompi ymmärtää. BASIC on eräs esimerkki lausekielistä.

Hiiri ks. Mouse

Huomautus ks. Remark

IEEE-488
Tämä on eräs standardiliitännöistä, jota käytetään esim. oheislaitteiden kytkemiseksi tietokoneeseen. IEEE-488 on monilta osiltaan samankaltainen - mutta ei täysin yhteensopiva - Centronics-tyyppisen rinnakkaisliitännän kanssa.

Information technology tietotekniikka
Tietotekniikka on kaikkea sitä, mikä liittyy elektroniikan käyttöön tietoliikenteessä ja tietojen käsittelyssä. Tietotekniikkaa ovat siis mm. tekstinkäsittely, tietoliikenne, teletietopalvelu (esim. Telset, Prestel, yms.), jne.

Initialise alustaa
Alustaminen tarkoittaa järjestelmän käynnistämistä tai tiettyjen arvojen antamista muuttujille ennen ohjelman suorituksen aloittamista. Alustamista voi olla esim. muuttujien määritteleminen kokonaisluvuiksi ennen ohjelman käynnistämistä, jne.

Input syöttö
Tämä nimike kattaa kaiken mitä tietokoneen muistiin syötetään näppäimistöltä, kasettinauhurilta, levyasemalta, sarjaliitännän kautta tai jostakin muusta syöttölähteestä.

Instruction käsky
Käsky on pyyntö tai komento, jolla tietokone saadaan suorittamaan haluttu toiminto. Ohjelma muodostuu joukosta peräkkäisiä käskyjä.

Instruction set käskyvalikoima
Käskyvalikoima koostuu kaikista niistä loogisista ja matemaattisista operaatioista, jotka mikroprosessori pystyy suorittamaan. Tämä tarkoittaa siis konekäskyvalikoimaa. Kukin lausekäsky (mukaan lukien symbolisen konekielen muistikkaat (mnemonics)) on voitava pelkistää käskyksi, jonka tietokoneen CPU tunnistaa. Yksi yksittäinen lausekielen käsky voi käynnistää suuren joukon CPU:n käskyvalikoiman konekäskyjä.

Integer number kokonaisluku
Kokonaisluku on luku, jossa ei ole desimaaliosaa eli desimaalipisteen oikealla puolella olevaa osaa. Vastakohtana on reaaliluku, jossa on kokonaisosa sekä desimaalipisteen oikealla puolella oleva desimaaliosa.

Integrated circuit integroitu piiri (t. IC-piiri)
IC-piiri koostuu elektroniikkapiireissä käytettävistä komponenteista, jotka on pienennetty miniatyyrikokoon ja rakennettu yhdelle piipalalle. Katso myös kohtaa "Chip".

Intelligent terminal älykäs pääte
Pääte, joka pystyy täyttämään tietokoneen syöttö- ja tulostusvaatimukset, mutta joka lisäksi pystyy itsenäisesti käsittelemään tietoja päätteen ollessa off line-tilassa (ei suorassa yhteydessä päätietokoneen kanssa).

Interactive vuorovaikutteinen
Vuorovaikutteisiksi ohjelmiksi kutsutaan yleensä ohjelmia, joissa laitteisto tulostaa esim. kuvaruutuun erilaisia käyttäjälle tarkoitettuja viestejä, joihin käyttäjä vastaa esim. näppäilemällä pyydetyt tiedot. Vuorovaikutteisia ohjelmia ovat niin avaruusaiheiset arkadipelit kuin opetusohjelmatkin, joissa käyttäjä vastaa tietokoneen esittämiin kysymyksiin. Käyttäjän toimet vaikuttavat "tosiaikaisesti" ohjelman käyttäytymiseen.

Interface liitäntä (t. liityntä)
Tapa, jolla tietokoneeseen päästään sisään ja ulos. Asiaa voidaan tarkastella teknisen näkökulman lisäksi käyttäjän kannalta eli millä tavalla käyttäjä "liittyy" tietokoneen toimintoihin. CPC 464:n liityntäkohtia ovat näppäimistö (tietojen syöttämiseksi) ja kuvaruutu (tietojen tulostamiseksi). Edelleen tietokoneen takaosassa on liitin, jolla oheislaitteita voidaan liittää CPC 464:ään.

Interpreter tulkki
Tulkki on tietokonejärjestelmän ohjelmiston osa, joka tulkitsee lausekielen CPU:n ymmärtämälle konekielelle. Se siis muuttaa näppäimistöltä näppäillyt BASIC-avainsanat tietokoneen omalle kielelle. BASIC-ohjelman tulkitseminen käsky käskyltä on hitaampaa, kuin jos koko ohjelma ensin käännetään konekielelle ja ajetaan sitten konekielisenä. (Katso kohtaa Compiler).

I/O
Input/Output = syöttö/tulostus.

Iteration iterointi
Iterointi on eräs tietokoneen perustoimintoja. Tietokone suorittaa kaikki tehtävät jakamalla ne yksinkertaisiksi tehtäviksi, jotka CPU pystyy suorittamaan. Voidakseen tehdä tämän tietokoneen on käytävä läpi lukuisia yksinkertaisia osatehtäviä, kunnes määritelty ehto täyttyy.

Joystick sauvaohjain
Syöttölaite, joka yleensä korvaa kohdistimen ohjausnäppäinten toiminnan. Näin esim. pelien pelaaminen on helpompaa ja nopeampaa.

k (t. K)
Pieni k-kirjain on lyhenne metrijärjestelmän etuliitteestä "kilo", ja se vastaa lukuarvoa 1000. Tietokoneiden yhteydessä k-kirjainta käytetään yleisesti kilotavun lyhenteenä. Todellisuudessa kilotavu on 1024 tavua. Se johtuu siitä, että 1 kilotavu on tarkasti 2 potenssiin 10 tavua, joka on desimaalimuodossa 1024 tavua. Aihetta käsitellään myös liitteessä II.

Kantaluku ks. Base

Kasetti ks. Cartridge ja Cassette

Katkaistu ks. Truncated

Kello ks. Clock

Keskusyksikkö ks. CPU

Keyboard näppäimistö
Näppäimistö käsittää aakkosnumeeriset näppäimet, joiden avulla käyttäjä voi näppäillä käskyjä tai muuta tietoa tietokoneeseen.

Keyword avainsana
Avainsana on sana, jonka käyttö on tarkasti rajattu tiettyä funktiota tai käskyä varten tietokoneen ohjelmassa tai kielessä.

Kiinteä laiteohjelmisto ks. Firmware

Kiinteä levy ks. Winchester

Kiintoluku ks. Fixed-point number

Kirjasinkiekkokirjoitin ks. Daisy-wheel printer

Kirjoitin ks. Printer

Kohdistin ks. Cursor

Kohdistimen ohjausnäppäimet ks. Cursor control keys

Kohina ks. Noise

Kokonaisluku ks. Integer number

Komento ks. Command

Koneen ja käyttäjän välinen liityntä ks. Man-machine interface

Koneenläheinen kieli ks. Low-level language

Konekieli (t. konekoodi) ks. Machine code, Machine language

Konekoodinen ks. Machine readable

Koodi ks. Code

Korjailu ks. Editing

Kuva-alkio ks. Pixel

Kuvaruutueditori ks. Screen editor

Käsky ks. Instruction

Käskyvalikoima ks. Instruction set

Kättely ks. Handshaking

Käyttöjärjestelmä ks. Operating system

Käänteinen puolalainen merkintätapa ks. Reverse Polish notation

Kääntäjäohjelma ks. Compiler

Laitteisto ks. Hardware

Lause ks. Statement

Lauseke ks. Expression

Lausekieli ks. High-level language

Least Significant Bit (LSB) vähiten merkitsevä bitti
Binääriluvun (katso liitettä II) vähiten merkitsevä bitti on lausekkeen äärimmäisenä oikealla oleva bitti.

Levy t. levyke ks. Disk tai Floppy disk

Levyasema ks. Disk drive

Light pen valokynä
Valokynä on eräs vaihtoehtoinen kynän muotoinen syöttöväline.

Liitäntä ks. Interface

Line number rivinumero
BASIC-ohjelmat ja eräät muunkieliset ohjelmat on järjestetty rivinumeroiden osoittamaan järjestykseen.

LISP
LISP on lyhenne sanoista LISt Processor Language. LISP on siis eräs tietokoneissa käytettävä lausekieli.

Liukuluku ks. Floating-point number

Logic logiikka
Logiikaksi kutsutaan elektroniikkakomponenteista rakennettua piiriä, joka suorittaa loogisia perusoperaatioita ja -toimintoja. Tietokoneen kaikki toiminnot perustuvat pohjimmiltaan juuri logiikkapiireihin.

LOGO
LOGO on grafiikkapainotteinen lausekieli, jota on alettu käyttää yhä enemmän kouluissa tietokoneopetuksen apuvälineenä lähinnä siksi, että se on helppo oppia.

Loop silmukka
Silmukaksi kutsutaan tiettyä ohjelman osaa, jota tietokone suorittaa yhä uudelleen ja uudelleen niin kauan kunnes tietty ehto täyttyy.

Low-level language koneenläheinen kieli
Tähän kieliryhmään kuuluu esim. symbolinen konekieli (assembly language). Se on ohjelmointikieli, jossa kukin käsky vastaa tietokoneen konekielistä käskyä.

LSI
LSI on lyhenne sanoista Large Scale Integration eli suurtiheysintegrointi. Se on eräs IC-piirien valmistustapa, jolla samalle piipalalle saadaan pakatuksi enemmän toimintoja pienempään tilaan.

Machine Code konekieli (t. konekoodi)
Machine language
Konekieli on ohjelmointikieltä, jota mikroprosessori ymmärtää suoraan, koska kaikki konekieliset käskyt on esitetty binäärimuodossa (ykkösinä ja nollina).

Machine readable konekoodinen (t. konekielinen)
Konekoodisella tiedolla tarkoitetaan sellaista tietoa, joka voidaan syöttää suoraan tietokoneeseen. Tieto on siis valmiiksi tietokoneen ymmärtämässä muodossa, eikä se edellytä esim. näppäimistöllä suoritettavaa lisätyötä haluttujen toimintojen käynnistämiseksi.

Man-machine interface koneen ja käyttäjän välinen liityntä
Tietokoneen ja sen käyttäjän välinen liityntäkohta eli näppäimistö, kuvaruutu, ääni, jne.

Matrix matriisi
Suorakaiteen muotoinen pistejoukko, joka muodostaa kuvaruudun merkkisolun tai pistematriisikirjoittimen kirjoituspään. Matriisi-nimitystä käytetään myös matemaattisten ja tietoteknisten taulukoiden nimenä.

Memory muisti
Muisti on "varasto", johon tietokone tallentaa tietoja. Muisti on järjestetty loogisesti sellaiseen muotoon, että kukin yksittäinen muistipaikka on erikseen osoitettavissa. CPC 464:ssä on kahdenlaista muistia eli RAM-muistia (Random Access Memory), johon tietoja voidaan tallentaa (kirjoittaa) ja josta tietoja voidaan lukea. Siitä tulee nimitys luku/kirjoitusmuisti. Toinen muistityyppi on ROM-muisti (Read Only Memory), josta tietoja voidaan lukea mutta johon tietoja ei enää voida kirjoittaa uudelleen toisessa muodossa. Siitä nimitys lukumuisti. Tietolevyt ja nauhat ovat massamuisteja (bulk memory), vaikkakin massamuistin merkitys on aikaa myöten kehittynyt tarkoittamaan muistia, jota tietokoneen keskusyksikkö voi osoittaa suoraan.

Memory map muistikartta
Muistikartta on eräänlainen sijoittelupiirustus, josta näkyy muistin eri osoitteet ja tietyille toiminnoille varatut muistipaikat, kuten kuvaruudun tai nauhurin ohjausjärjestelmälle varatut muistipaikat.

Menu valikko
Valikossa on lueteltu eri vaihtoehdot, jotka tietokoneella voi suorittaa kyseisessä valintatilanteessa. Valinnan suorittaa koneen käyttäjä.

Merkki ks. Character

Merkkijono ks. Character string

Merkkijonomuuttuja ks. String variable

Merkkisolu ks. Character cell

Merkkivalikoima ks. Character set

Microprocessor mikroprosessori (t. mikrosuoritin)
Mikroprosessori on mikrotietokoneen keskeisin integroitu piiri, joka suorittaa BASIC-tulkilta saamansa käskyt voidakseen ohjata mm. eri tulostuslaitteita.

Mnemonic muistikas ks. Assembly language

Modem modeemi
Modeemi tulee sanoista MOdulaattori/DEModulaattori, jolla tietokoneen syöttö/tulostus (I/O) kytketään puhelinlinjaan tai johonkin muuhun vastaavaan sarjamuotoiseen tiedonsiirtovälineeseen - mukaan lukien valokaapelit. Katso myös kohtaa Acoustic coupler.

Moduuli ks. Cartridge

Monitor monitori
Monitori-sanalla on kaksi merkitystä. Toinen tarkoittaa tietokonejärjestelmän näyttöyksikköä ja toinen konekielistä ohjelmaa, jolla voidaan tutkia tietokoneen konekielen toimintaa.

Most Significant Bit (MSB) eniten merkitsevä bitti
Binääriluvun eniten merkitsevä bitti on lausekkeen vasemmanpuoleisin bitti.

Mouse hiiri
Hiiri on pöydällä liikuteltava tietokoneeseen kytketty laite, jolla yleensä ohjataan kohdistimen liikkeitä kuvaruudun alueella. Alunperin hiiri suunniteltiin pienentämään tietokonenäppäimistöön kohdistuvaa pelkoa ja antamaan ohjelmista "käyttäjäystävällisempi" vaikutelma.

Muisti ks. Memory

Muistikartta ks. Memory map

Muotovirhe ks. Syntax error

Muuttuja ks. Variable

Network verkko
Kaksi tai useampia tietokoneita voidaan yhdistää toisiinsa, jolloin ne voivat vaihtaa keskenään tietoja. Tietokoneet voidaan kytkeä tällä tavalla verkkoon joko suoraan kaapeleilla tai käyttäen modeemeja. (Ei saa sekoittaa sähköverkkoon, josta tietokoneet saavat käyttöjännitteensä 220 V/50 Hz).

Nibble puolitavu
Puolitavu on neljä bittiä 8-bittisestä tavusta. Kukin heksadesimaalinumeroista lausekkeessa &F6 vastaa yhtä puolitavua.

Noise kohina
CPC 464:n äänigeneraattorin ominaisuuksiin kuuluu mahdollisuus muodostaa säädettävää valkoista kohinaa SOUND-käskyllä. Näin saadaan muodostetuksi esim. räjähdysääniä kuvaavia tehosteita.

Numeric keypad numeronäppäimistö
Numeronäppäimistö on näppäimistön se osa, johon numeronäppäimet on ryhmitelty helpottamaan numerotietojen syöttämistä. CPC 464:n tapauksessa numeronäppäimistöstä on se lisäetu, että käyttäjä voi halutessaan määritellä numeronäppäimille uusia toimintoja. (Katso myös kohtaa Function key).

Näppäimistö ks. Keyboard

OCR
OCR on lyhenne sanoista Optical Character Recognition eli optinen merkin tunnistus. Se on tapa lukea OCR-merkkejä optisella lukijalla ja muuttaa ne suoraan tietokoneen ymmärtämiksi tiedoiksi.

Octal oktaalimuoto
Oktaalijärjestelmän lukujen kantaluku on 8, jossa kukin numero (0 - 7) koostuu kolmesta bitistä.

Off line erillinen
Off line tarkoittaa tilaa, jossa tietokoneen oheislaite - tavallisesti näyttöpääte tai kirjoitin - ei ole parhaillaan kytkettynä keskustietokoneeseen, jolloin myöskään keskustietokone ei pääse yhteyteen oheislaitteen kanssa.

Oheislaite ks. Peripheral

Ohjelma ks. Program

Ohjelmanäppäin ks. Soft key

Ohjelmasuunnittelu ks. Software engineering

Ohjelmavirhe ks. Bug

Ohjelmisto ks. Software

Ohjelmointikieli ks. Programming language

On line suoraan kytketty
Tila, jossa esim. oheislaite on suoraan kytkettynä tietokoneeseen. Vastakohta Off line.

Operating system käyttöjärjestelmä
Käyttöjärjestelmä on tietokoneohjelmisto, joka huolehtii tietokoneen eri toimintojen ajastuksesta sallien eri toimintojen tapahtumisen oikealla hetkellä.

Osoite ks. Address

Output tulostus, tuloste
Nimike pitää sisällään kaiken sen, mitä tietokoneesta saadaan laskentatoimenpiteiden tuloksena.

Operator operaattori
Operaattoriksi kutsutaan aritmeettisen lausekkeen sitä osaa, joka saa aikaan sen, että yksi luku operoi toista lukua eli suorittaa sille halutun operaation. Operaattoreita ovat esim. merkit + - * / jne.

Overwrite päällekirjoitus
Päällekirjoitus tarkoittaa uuden tiedon kirjoittamista vanhan tiedon päälle siten, että vanha tieto häviää.

Paddle peliohjain
Peliohjaimella tarkoitetaan sellaista ohjainta, jossa ohjaus tapahtuu säätönuppia kääntämällä. Erotuksena sauvaohjain, jossa ohjaus tapahtuu ohjaussauvaa liikuttelemalla.

Paperikopio ks. Hard copy

Paperware paperitavara
Paperitavara on toinen nimike tietokoneen tulosteen paperikopiolle (hard copy). Joskus myös joitakin tietokoneita voidaan kutsua "paperitavaraksi", jos ne julkaistaan jo ennen kuin niiden kehitystyö on suoritettu loppuun.

Parallel interface rinnakkaisliitäntä
CPC 464:ssä on rinnakkaismallinen kirjoitinliitäntä (Centronics). Tämä tarkoittaa sitä, että kukin tietokoneen väylän datalinja kytketään vastaavaan kirjoittimen tulolinjaan. Tietoja pystytään siirtämään nopeammin rinnakkaisliitännän kautta sarjaliitäntään varrattuna, koska sarjaliitännässä kukin tavu on ensin muotoiltava (format) ja tahdistettava tahdistussignaalin kanssa.

PASCAL
PASCAL on eräs lausekielinen lohkorakenteinen ohjelmointikieli, joka täytyy kääntää (compile) ennen ohjelman suoritusta. Tästä syystä PASCAL-ohjelmalla tehdyt ohjelmat toimivat hyvin nopeasti. PASCAL on yleensä seuraava kieli, johon innokas BASIC-harrastaja haluaa perehtyä.

PEEK
PEEK on BASIC-funktio, jolla päästään "kurkistamaan" suoraan tietokoneen muistipaikkaan ja jonka avulla saadaan selville määritellyn muistipaikan sisältö.

Peliohjain ks. Paddle

Peripheral oheislaite
Oheislaitteita ovat kirjoittimet, modeemit, sauva- ja peliohjaimet, levyasemat eli kaikki sellaiset laitteet, jotka voidaan liittää teitokoneeseen täydentämään sen toimintamahdollisuuksia.

Piirturi ks. Plotter

Pino ks. Stack

Pistematriisi ks. Dot matrix

Pixel kuva-alkio
Kuva-alkioksi nimitetään pienintä yksittäistä kuvaruudun osaa, jota voidaan ohjata laitteistolla.

Plotter piirturi
Piirturi on tulostuslaite, joka piirtää pitkiä viivoja käyttäen usein erivärisiä kyniä kirjoittimen tulostustavasta poiketen, jossa tulostusjälki yleensä muodostuu iskun vaikutuksesta. Piirturia käytetään teknisten ja graafisten piirustusten tulostukseen.

POKE
POKE on BASIC-käsky, jota käytetään halutun arvon tallennukseen määriteltyyn muistipaikkaan. Katso POKE-käskyn kuvausta luvusta 8.

Port portti
Portti on tietojen syöttöön tai tulostukseen käytettävän liitännän erikseen osoitettavissa oleva kohta.

Portability siirrettävyys
Siirrettävyydellä tarkoitetaan mahdollisuutta käyttää ohjelmistoja erimerkkisissä tietokoneissa. Tämä on yleensä Digital Researchin CP/M-järjestelmän kaltaisten yhteensopivien käyttöjärjestelmien ansiota.

Primer alkeisharjoitus
Alkeisharjoitukseksi kutsutaan tiettyä annosta peruskäskyjä ja ohjeita, joilla käyttäjälle esitellään määrättyä tietokoneen ominaisuutta.

Printer kirjoitin
Kirjoittimeksi kutsutaan mitä hyvänsä tulostuslaitetta, jolla saadaan paperikopio tulostettavasta tekstistä.

Program ohjelma
(Tietokoneiden yhteydessä Program eikä Programme). Ohjelmaksi kutsutaan tiettyyn järjestykseen peräkkäin asetettuja käskyjä, joilla tietokone saadaan suorittamaan haluttu tehtävä. Ohjelma voi olla varsin yksinkertainen konekielinen rutiini tai monipuolinen sovellutusohjelma, kuten tekstinkäsittelyohjelma.

Programming language ohjelmointikieli
Ohjelmointikieli on eräänlainen apuväline, jolla ohjelma kirjoitetaan. Kullakin ohjelmointikielellä on omat tiukat sääntönsä siitä, millä tavalla avainsanoja ja lukuja käytetään ja missä järjestyksessä ne suoritetaan.

PROM
PROM on lyhenne sanoista Programmable Read Only Memory eli ohjelmoitava lukumuisti. Se on siis integroitu muistipiiri, jota ei voida pyyhkiä sen jälkeen kun siihen on kerran tallennettu tietoja. (Katso myös kohtaa EPROM).

Prompt viesti
Viestiksi kutsutaan symbolia tai ilmoitusta, joka ilmestyy kuvaruutuun. Viesti kehottaa käyttäjää toimimaan tietokoneen haluamalla tavalla eli vastaamaan tietokoneen esittämään kysymykseen tai näppäilemään tietokoneen tarvitsemia tietoja. BASIC-kielessä käytetään kysymysmerkkiä **?** haluttaessa käyttäjän näppäilevän tietoja tai sanaa **Ready** tietokoneen odottaessa näppäimistöltä annettavaa käskyä.

PSU
PSU on lyhenne sanoista Power Supply Unit eli virtalähde. Virtalähde on se yksikkö, joka muuttaa verkkojännitteen (220 V/50 Hz) tietokoneen (ja oheislaitteiden) toiminnan kannalta tarpeellisiksi jännitteiksi.

Puolitavu ks. Nibble

Puheentunnistus ks. Speech recognition

Puhesynteesi ks. Speech synthesis

Puskuri ks. Buffer

Päällekirjoitus ks. Overwrite

Pääte ks. Terminal

QWERTY keyboard kirjoituskonenäppäimistö
Tällä nimityksellä halutaan korostaa sitä seikkaa, että tietokoneen näppäinten sijoitus on tavanomaisen kirjoituskoneen näppäimistön kaltainen.

Rakenteinen ohjelmointi ks. Structured programming

RAM
RAM on lyhenne sanoista Random Access Memory eli luku/kirjoitusmuisti. RAM-muistiin voidaan tallentaa tietoja ja siitä voidaan lukea tietoja käyttäen tietokoneen sisäisiä piirejä ohjelman normaalin suorituksen aikana.

Random number satunnaisluku
Satunnaisluvuksi kutsutaan tietokoneohjelman muodostamaa lukua, jonka arvo on satunnainen (ei toistuva) ja jota ei pysty ennalta aavistamaan.

Raster rasteri
Rasteri on menetelmä, jolla kuvioita "kirjoitetaan" kuvaruutuun. Kuviot muodostuvat joukosta vaakasuunnassa pyyhkäistäviä juovia (rasteripyyhkäisy).

Real number reaaliluku
Reaaliluku on luku, jossa on sekä kokonais- että desimaaliosa. Toisin sanoen luvun desimaalipisteen molemmilla puolilla olevia osia käytetään. Muuttujaa voidaan silti pitää reaalilukuna siitä huolimatta, vaikka se varaisi vain kokonaisluvun paikan tutkittaessa muuttujan arvoa jossakin tietyssä ohjelman suorituksen kohdassa.

Real time tosiaika (t. reaaliaika)
Tosiaikaisia tapahtumia ovat sellaiset, jotka tapahtuvat juuri silmiesi edessä vastakohtana niille, jotka ilmestyvät näkyviin vasta sitten kun ilmiöt tuottanut prosessi on päättynyt.

Recursive rekursiivinen (t. itseensä palautuva)
Rekursiiviseksi eli itseensä palautuvaksi kutsutaan joukkoa ohjelmaan tai rutiiniin kuuluvia toistuvia vaiheita, joissa kunkin toistetun tapahtumajakson tulos ohjataan vaikuttamaan aikaisempaan tulokseen.

Refresh virkistys
Virkistyksellä tarkoitetaan joko näyttölaitteen kuvaruudulla tai tietokoneen muistissa olevien tietojen päivittämistä. Sen ei välttämättä tarvitse olla aikaisempia tietoja tuhoava prosessi vaan pikemminkin sillä vahvistetaan sitä mikä muistissa tai kuvaruudulla oli ennen virkistystä.

Register rekisteri
Rekisteri on keskusyksikössä (CPU) oleva väliaikainen muistipaikka, jota käytetään tietojen väliaikaiseen tallennukseen.

Remark huomautus
Huomautuksiksi kutsutaan ohjelmiin REM-käskyllä sijoitettuja huomatus- tai selityslausekkeita, joilla ohjelmoijaa muistutetaan kunkin ohjelman osan suorittamasta tehtävästä. Samoin ohjelman alkuun voidaan lisätä REM-lausekkeella kyseisen ohjelmaversion päiväystiedot.

Reserved word varattu sana
Varattu sana on sana, jolla on erityinen merkitys tietokoneohjelmassa, eikä sitä voida käyttää muuten kuin sen tarkasti määritellyssä muodossa. BASIC ei esimerkiksi hyväksy sanaa NEW muuttujaksi, koska se on jo "varattu" muihin tarkoituksiin.

Resolution erottelukyky
Kyky erottaa kohta, jossa yksi näytön kuva-alkio päättyy ja toinen alkaa. Väljästi tulkittuna tarkoittaa myös tietokoneen kykyä suorittaa aritmeettisia toimenpiteitä suurilla luvuilla.

Reverse Polish notation (RPN) käänteinen puolalainen merkintätapa
Tämä on eräiden laskinvalmistajien suosima aritmeettisten operaatioiden merkintätapa, jossa operaattorit (+, -, x, /) laitetaan niiden lukujen jälkeen, joihin operaatio kohdistuu (siis operaattorit operandien jälkeen).

RF modulator suurtaajuusmodulaattori
Suurtaajuusmodulaattori muuttaa tietokoneesta saatavan videosignaalin suurtaajuussignaaliksi ja "lähettää" sen tavallisen television antenniliitäntään.

Rinnakkaisliitäntä ks. Parallel interface

Rivinumero ks. Line number

ROM
ROM on lyhenne sanoista Read Only Memory eli lukumuisti. Yleensä sillä tarkoitetaan puolijohdemuistia, johon kerran kirjoitettua (tallennettua) tietoa ei saa pyyhityksi pois eikä siihen voi kirjoittaa uutta tietoa.

Routine rutiini
Rutiiniksi kutsutaan sellaista ohjelman osaa, joka suorittaa "rutiinitehtävän". Se on siis eräänlainen aliohjelma, joka voi sijaita pääohjelmassa, tai se voi olla erillisellä moduulilla, jolloin sitä voidaan käyttää erilaisissa sovellutusohjelmissa. Rutiinina voidaan pitää esimerkiksi ohjelmaa, joka muodostaa järjestelmän kellosta 12-tuntisen kellonäytön.

RS232C
RS323C on erikoisstandardi, joka koskee tietoliikenteen sarjaliitäntöjä. Tietoliikenneyhteyden molemmissa päissä olevat RS232-liitäntää käyttävät laitteet on "konfiguroitava" tiettyyn käyttökuntoon ennen kuin tiedonsiirto onnistuu. Centronics-tyyppinen rinnakkaisliitäntä on kätevämpi, koska sitä käyttävien laitteiden välinen kytkentä on tarkasti standardoitu.

Rutiini ks. Routine

Sarjaliitäntä ks. Serial interface

Satunnaisluku ks. Random number

Sauvaohjain ks. Joystick

Screen editor kuvaruutueditori
Kuvaruutueditoriksi kutsutaan teksti- tai ohjelmaeditoria, joka suo mahdollisuuden siirtää kohdistimen mihin kuvaruudun kohtaan hyvänsä, jolloin kuvaruudun eri kohdissa olevia merkkejä voidaan muuttaa.

Scrolling vieritys
Vieritykseksi kutsutaan tapaa, jolla näyttöä vieritetään ylöspäin näytön täyttyessä viimeistä riviä myöten, jolloin näyttöä täytyy siirtää ylöspäin, jotta seuraava rivi voitaisiin sijoittaa alimmaksi kuvaruutuun näkyviin.

Seikkailupeli ks. Adventure game

Separator erotin (t. rajamerkki)
Erotin toimii samalla tavalla kuin rajamerkkikin (delimiter) eli merkitsee rajattujen sanojen ja muiden ohjelma- tai tietoalkioiden välisen rajakohdan.

Serial interface sarjaliitäntä
Vaikka sarjaliitännällä melkein aina tarkoitetaankin RS232-liitäntää, on tietokoneen tietojen perättäissiirtoon olemassa myös muita vakiintuneita sarjaliitäntöjä.

Siirrettävyys ks. Portability

Silmukka ks. Loop

Simulation simulointi (t. jäljittely)
Simuloinniksi kutsutaan tietokoneen avulla suoritettavaa todellisen vuorovaikutteisen tapahtuman jäljittelyä. Esimerkkeinä lentosimulaattorit, ajosimulaattorit, jne.

Siru ks. Chip

Soft key ohjelmanäppäin (t. "pehmonäppäin")
Katso kohtaa UDK (User Defined Key eli käyttäjän määriteltävissä olevat näppäimet).

Software ohjelmisto
Ohjelmisto pitää sisällään kaikki ohjelmat. (Vrt. Hardware eli laitteisto).

Software engineering ohjelmasuunnittelu
Ohjelmasuunnittelulla tarkoitetaan tavoitteellista ja harkittua tapaa suunnitella tietokoneohjelmia. Tällä on haluttu erottaa ohjelmien tekotapa, joka perustuu enemmänkin sattumanvaraisiin menetelmiin.

Sound generator äänigeneraattori
Äänigeneraattoriksi kutsutaan sitä tietokoneen osaa, joka muodostaa äänet ja kohinan. Se voi olla joko laitteistoa tai ohjelmistoa.

Sovellus- tai sovellutusohjelma ks. Applications program

Speech recognition puheentunnistus
Puheentunnistus tarkoittaa puhuttujen sanojen muuttamista konekoodisiksi käskyiksi.

Speech synthesis puhesynteesi (t. puheenmuodostus)
Puhesynteesillä tarkoitetaan keinotekoisen puheäänen muodostamista tietokonelaitteiston ja -ohjelmiston avulla.

Spreadsheet taulukkolaskentaohjelma
Ohjelma, joka suo mahdollisuuden lukujen syöttämiseen ja aritmeettiseen käsittelyyn riveittäin ja sarakkeittain. Yhden syöttötiedon muuttaminen saa aikaan sen, että kaikki siihen liittyvät laskutoimitukset suoritetaan uudelleen ja näin saadaan päivitetty tulos.

Sprite
Kuvaruutumerkki, joka pystyy liikkumaan vapaasti koko näytön alueella. Se on muodostettu erityislaitteistolla tai -ohjelmistolla, ja se voi ilmestyä näkyviin ja kadota kuvaruudusta näennäisen sattumanvaraisesti.

Stack pino
Pinoksi kutsutaan muistialuetta, joka on varattu tietojen "pinoamiseen". Yleisin pinomuisti on sellainen, josta viimeksi ("päällimmäiseksi") pinoon laitettu tieto saadaan ensimmäiseksi uudelleen käyttöön.

Statement lause (t. käsky)
Tietokoneohjelmassa oleva käsky tai useammasta käskystä koostuva lause.

Stream tulostusreitti
Tulostusreitiksi kutsutaan reittiä, jota käytetään tulostettaessa tietoja tietokoneelta esim. kuvaruutuun, kirjoittimelle tai kasettinauhurille.

String variable merkkijonomuuttuja
Merkkijonomuuttujaksi kutsutaan sellaista datatyyppiä, joka koostuu erilaisista merkeistä, ja jota ei voida pitää numeromuuttujana. Se voi myös olla puhtaasti numeromuotoinen, mutta sitä ei käsitellä numeromuuttujana ellei sitä erikseen muuteta vastaavaksi numeromuuttujaksi sopivalla käskyllä.

Structured programming rakenteinen ohjelmointi
Tämä on looginen ja harkittu ohjelmointitapa, jonka tuloksena syntyy ohjelmia, jotka kulkevat "ylhäältä alas" selvästi erottuvien ohjelmavaiheiden kautta.

Subroutine aliohjelma
Katso kohtaa Routine.

Suoraan kytketty ks. On line

Suurtaajuusmodulaattori ks. RF modulator

Syntax error muotovirhe
Muotovirhe tarkoittaa tilannetta, jossa ohjelman muotosääntöjä on rikottu käyttämällä avainsanoja tai muuttujia väärin. Tällöin BASIC tulostaa kuvaruutuun virheilmoituksen. Katso liitettä VIII.

Syöttö ks. Input

Syöttö/tulostus ks. I/O

Taulukkolaskentaohjelma ks. Spreadsheet

Tavu ks. Byte

Tekoäly ks. Artificial intelligence

Terminal pääte
Päätteeksi kutsutaan näppäimistöllä varustettua syöttölaitetta, jossa tulostuslaitteena on joko kuvaruutu tai kaukokirjoitin.

Tiedosto ks. File

Tietojen keräys ks. Data capture

Tietokanta ks. Database

Tietokoneavusteinen opetus ks. CAE

Tietokoneavusteinen suunnittelu ks. CAD

Tietokonesukupolvi ks. Computer generation

Tietolevy ks. Disk tai Floppy Disk

Tietotekniikka ks. Information Technology

Toimintonäppäin ks. Function key

Toimitusohjelma ks. Editor

Tosiaika ks. Real time

Totuustaulukko ks. Truth table

Truncated katkaistu
Luku tai merkkijono, jota on lyhennetty poistamalla merkkejä sen alusta tai lopusta. Tarkoituksellinen katkaisu voi tarkoittaa arvon pyöristämistä ja tahaton tai koneen suorittama katkaisu tarkoittaa sitä, että ylimääräiset merkit yksinkertaisesti hylätään, jotta luku tai merkkijono mahtuu käytettävissä olevaan tilaan.

Truth table totuustaulukko
Loogisten operaatioiden tulokset ovat joko "tosia" tai "epätosia". Tietokone tulkitsee nämä joko ykkösiksi tai nolliksi, ja totuustaulukossa on lueteltu loogisten operaatioiden (esim. IF A>B THEN C) kaikki mahdolliset tulokset.

Tulkki ks. Interpreter

Tulostus ks. Output

Tulostusreitti ks. Stream

Tyhmä pääte ks. Dumb terminal

UDK
UDK on lyhenne sanoista User Defined Keys eli käyttäjän määriteltävissä olevat näppäimet. CPC 464:ssä on jopa 32 näppäintä, jotka voidaan määritellä uudelleen suorittamaan erilaisia tehtäviä. Kullekin näppäimelle ohjelmoitava käskymerkkijono voi olla pituudeltaan enintään 32 merkkiä.

Unsigned number etumerkitön luku
Luku, jossa ei ole etuliitettä sen merkiksi, onko luku positiivinen vaiko negatiivinen.

Utility program apuohjelma
Apuohjelmaksi kutsutaan mitä hyvänsä täydellistä ohjelmaa, joka suorittaa tietyn yhteisen tehtävän kuten tietojen lajittelun tai tiedostojen kopioinnin.

Uusi lukutaito ks. Computer literacy

Valikko ks. Menu

Valokynä ks. Light pen

Varattu sana ks. Reserved word

Variable muuttuja
Muuttujaksi kutsutaan sellaista tietokoneohjelman osaa, joka voidaan tunnistaa omalla nimellään, mutta jonka arvoa voidaan muuttaa ohjelman suorituksen aikana.

Vastaanottaa lataustietoja ks. Download

Verkko ks. Network

Veräjä ks. Gate

Vieritys ks. Scrolling

Viesti ks. Prompt

Viidennen sukupolven tietokoneet ks. Fifth generation computers

Viivakoodi ks. Bar code

Virheiden poisto ks. Debugging

Virkistys ks. Refresh

Vuokaavio ks. Flowchart

Vuorovaikutteinen ks. Interactive

Vähiten merkitsevä bitti ks. Least Significant Bit

Väylä ks. Bus

Winchester kiinteä levy (t. kovalevy)
Kiinteä levy on yleensä tietokoneeseen kiinteästi asennettu tiedontallennusväline, johon mahtuu huomattavasti enemmän tietoa kuin tietolevylle. Tallennuskapasiteetti on suuruusluokkaa 10 - 20 MB (= megatavua).

Älykäs pääte ks. Intelligent terminal

Älytön pääte ks. Dumb terminal

Äänigeneraattori ks. Sound generator

BITIT JA TAVUT

Olemme tottuneet jokapäiväisessä elämässä käyttäämään kymmenjärjestelmän eli desimaalijärjestelmän lukuja, joiden kantaluku on 10. Siinä kaikki luvut ilmaistaan numeroiden 0 - 9 avulla. Binäärijärjestelmässä on kuitenkin käytettävissä vain kaksi numeroa: 0 ja 1, joita kutsutaan biteiksi.

Binäärijärjestelmän luvut muodostetaan kakkosen potensseista seuraavasti:

$2^0 = 1$
$2^1 = 2 = 2 = 2(2^0)$
$2^2 = 4 = 2 \times 2 = 2(2^1)$
$2^3 = 8 = 2 \times 2 \times 2 = 2(2^2)$
$2^4 = 16 = 2 \times 2 \times 2 \times 2 = 2(2^3)$
$2^5 = 32...$
...

Näin siis esimerkiksi desimaaliluku 7 on binäärimuodossa 00111, jos käytämme 5-bittistä merkintätapaa. Etunollat eivät ole merkitseviä, vaan ykköset osoittavat, mitä kakkosen potensseja binäärilukuun sisältyy.

Toinen esimerkki:

2^4	2^3	2^2	2^1	2^0
1	0	0	1	1
(16	0	0	2	1) = 19

Yksi tavu koostuu kahdeksasta bitistä. Suurin tavuun mahtuva luku on binäärimuodossa 1111111 eli desimaalimuodossa 255. Yhdellä tavulla saadaan siis esitetyksi 256 eri vaihtoehtoa, koska myös 0000000 on täysin käyttökelpoinen tieto tietokoneelle.

Tietokone käsittelee tietoja 8 bitin kerrannaisina. 256 ei ole mikään kovin suuri luku, joten tietokoneissa käytetään kahta tavua muistin osoittamiseksi, jotta muistin käsittely kävisi kätevästi päinsä. Muisti on järjestetty taulukon muotoon, jossa on vaaka- ja pystyosoitteet, joilla taulukon eri alkiot saadaan määritellyiksi.

0	1	2	3	4	5	6	7	8	9
1									
2									
3									
4									
5				1		1			
6									
7									
8									
9									

Edellisellä sivulla olevassa taulukossa on 10 x 10 alkiota, ja siinä käytetään välillä 1 - 9 olevia muistiosoitenumeroita. Paikkaan 3,5 tallennettu tietoalkio on "1" samoin kuin paikkaan 5,5 tallennettu tietokin.

Täten 256 x 256 alkiosta koostuva binäärimuotoinen muistimatriisi pystyy käsittelemään 65536 yksittäistä muistipaikkaa käytettäessä 8-bittisiä osoitteita matriisin pysty- ja vaakariveillä. Niinpä binääriluvuilla 0 ja 1 pystytään yksilöimään jo 65536 muistipaikkaa.

Yksi tavu piti sisällään 8 bittiä ja seuraava porras on 1 kilotavu, johon mahtuu 1024 tavua. Tarkasti ottaen 1 kilotavu on 2^{10} tavua, mutta käytännössä riittää kun muistaa, että 1 kilotavu on noin 1000 tavua. Jos tietokoneen muistikapasiteetiksi ilmoitetaan 64 kilotavua, on se siis tarkkaan ottaen 65536 tavua (64 x 1024 tavua).

Oikeastaan sinun ei tarvitse paljoakaan välittää binääriluvuista ainakaan aluksi, sillä BASIC muuttaa tutut desimaaliluvut tietokoneen ymmärtämään binäärimuotoon. Sinusta voi tulla aivan hyvin vaikka kuinka taitava ohjelmoija, vaikka et kokonaan ymmärtäisikään binäärijärjestelmää. Binäärijärjestelmän ymmärtäminen helpottaa kuitenkin tiettyjen laskentatapahtumien ja lukujen ymmärtämistä varsinkin opintojesi pitemmälle edistyessä.

HEKSADESIMAALIMUOTO

Binäärijärjestelmän kiistattomista eduista huolimatta sillä on myös mm. se haitta, ettei binäärilukujen arvoa näe helposti yhdellä silmäyksellä. Binäärijärjestelmästä on kehitetty muitakin lukujärjestelmiä, ja eräs sellainen tietokoneiden yhteydessä laajalti käytettty on nimeltään heksadesimaalijärjestelmä. Siinä kantaluku on 16, ja merkintätapa on seuraava:

Desimaalimuoto:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Heksadesimaalimuoto:	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Heksadesimaalijärjestelmässä tavun 8 bittiä voidaan jakaa kahdeksi neljän bitin lohkoksi, sillä 15 on 4-bittinen luku 1111 binäärimuodossa. Ensimmäinen lohko osoittaa täydellisten "15":n yksikköjen lukumäärän ja toinen lohko ilmaisee "jäännöksen". Seuraavalla sivulla olevassa taulukossa on desimaalilukujen 0 - 16 vastaavat binääri- ja heksadesimaaliarvot.

Desimaalimuoto	Binäärimuoto	Heksadesimaalimuoto
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	10000	10

8-bittinen luku 11010110 voidaan jakaa kahdeksi 4-bittiseksi luvuksi eli puolitavuiksi (Hex D6). Heksadesimaaliluvut on merkitty kauttaaltaan tässä käsikirjassa symbolilla & esim &D6. Symbolisen konekielen (assembly language) ohjelmoijat käyttävät laajalti heksadesimaalimuotoa. Symbolinen konekieli on puolestaan ohjelmointikieli, jolla päästään hyvin lähelle puhdasta konekieltä, koska symbolisessa konekielessä käytetään yksinkertaisia "muistikkaita" (mnemonics), joilla määritellään varsinaiset konekieliset "luvut".

Heksadesimaaliluvun arvoa määriteltäessä on ensin laskettava ensimmäisen numeron arvo, jotta saataisiin selville kuusientoista lukumäärä lopullisessa luvussa. Tämän jälkeen heksadesimaaliluvun toiseksi "puolikkaaksi" nimitettävä jäännös lisätään edelliseen tulokseen, jolloin saadaan koko luvun desimaaliarvo. Saattaa tuntua houkuttelevalta ajatella, että heksadesimaaliluku &D6 olisi 13 + 6 tai 136, mutta se on todellisuudessa (13 x 16) + (6) = 214.

Itse asiassa käytät samaa menetelmää lukiessasi desimaalijärjestelmän (eli kymmenjärjestelmän) lukua. Esimerkiksi 89 on (8 x 10) + (9). On vain paljon helpompaa kertoa kymmenellä kuin kuudellatoista, ellei sinulla ole siitä ennestään kokemusta.

Jos olet pysynyt kärryillä tähän asti menemättä päästäsi pyörälle, alat olla varsin vahvasti jyvällä tietokoneen toiminnan periaatteista. Eihän tietokone loppujen lopuksi ole muuta kuin erittäin nopea laskukone, jolla on vielä varsin hyvä muisti (tietyin rajoituksin).

Emme ole voineet käsitellä asioita kovinkaan syvällisesti tämän käsikirjan puitteissa, joten kannattaa vierailla lähimmässä kirjakaupassa, josta varmasti löytyy runsaasti tietokoneita käsittelevää kirjallisuutta. Tuoreinta tietoa halutessasi saat sitä alan lehdistä.

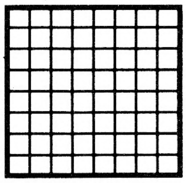
III.1 ASCII-KOODIT

Seuraavaan taulukkoon on kerätty ASCII-merkkivalikoima vastaavine desimaali-, oktaali- ja heksadesimaalimuotoineen. Seuraavilla sivuilla on myös esitetty kaikki CPC 464:n merkkisolut 8 x 8 matriiseina.

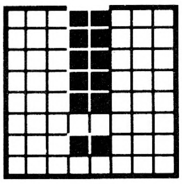
DEC	OCTAL	HEX	ASCII characters	DEC	OCTAL	HEX	ASCII	DEC	OCTAL	HEX	ASCII
0	000	00	NUL ([CTRL]A)	50	062	32	2	100	144	64	d
1	001	01	SOH ([CTRL]B)	51	063	33	3	101	145	65	e
2	002	02	STX ([CTRL]C)	52	064	34	4	102	146	66	f
3	003	03	ETX ([CTRL]C)	53	065	35	5	103	147	67	g
4	004	04	EOT ([CTRL]D)	54	066	36	6	104	150	68	h
5	005	05	ENQ ([CTRL]E)	55	067	37	7	105	151	69	i
6	006	06	ACK ([CTRL]F)	56	070	38	8	106	152	6A	j
7	007	07	BEL ([CTRL]G)	57	071	39	9	107	153	6B	k
8	010	08	BS ([CTRL]H)	58	072	3A	:	108	154	6C	l
9	011	09	HT ([CTRL]I)	59	073	3B	;	109	155	6D	m
10	012	0A	LF ([CTRL]J)	60	074	3C	<	110	156	6E	n
11	013	0B	VT ([CTRL]K)	61	075	3D	=	111	157	6F	o
12	014	0C	FF ([CTRL]L)	62	076	3E	>	112	160	70	p
13	015	0D	CR ([CTRL]M)	63	077	3F	?	113	161	71	q
14	016	0E	SO ([CTRL]N)	64	100	40	@	114	162	72	r
15	017	0F	SI ([CTRL]O)	65	101	41	A	115	163	73	s
16	020	10	DLE ([CTRL]P)	66	102	42	B	116	164	74	t
17	021	11	DC1 ([CTRL]Q)	67	103	43	C	117	165	75	u
18	022	12	DC2 ([CTRL]R)	68	104	44	D	118	166	76	v
19	023	13	DC3 ([CTRL]S)	69	105	45	E	119	167	77	w
20	024	14	DC4 ([CTRL]T)	70	106	46	F	120	170	78	x
21	025	15	NAK ([CTRL]U)	71	107	47	G	121	171	79	y
22	026	16	SYN ([CTRL]V)	72	110	48	H	122	172	7A	z
23	027	17	ETB ([CTRL]W)	73	111	49	I	123	173	7B	{
24	030	18	CAN ([CTRL]X)	74	112	4A	J	124	174	7C	|
25	031	19	EM ([CTRL]Y)	75	113	4B	K	125	175	7D	}
26	032	1A	SUB ([CTRL]Z)	76	114	4C	L	126	176	7E	~
27	033	1B	ESC	77	115	4D	M				
28	034	1C	FS	78	116	4E	N				
29	035	1D	GS	79	117	4F	O				
30	036	1E	RS	80	120	50	P				
31	037	1F	US	81	121	51	Q				
32	040	20	SP	82	122	52	R				
33	041	21	!	83	123	53	S				
34	042	22	"	84	124	54	T				
35	043	23	#	85	125	55	U				
36	044	24	$	86	126	56	V				
37	045	25	%	87	127	57	W				
38	046	26	&	88	130	58	X				
39	047	27	'	89	131	59	Y				
40	050	28	(	90	132	5A	Z				
41	051	29	)	91	133	5B	[				
42	052	2A	*	92	134	5C	\				
43	053	2B	+	93	135	5D	]				
44	054	2C	,	94	136	5E	↑				
45	055	2D	-	95	137	5F	_				
46	056	2E	.	96	140	60	`				
47	057	2F	/	97	141	61	a				
48	060	30	Ø	98	142	62	b				
49	061	31	1	99	143	63	c				

III.2 CPC 464:N MERKKIVALIKOIMA

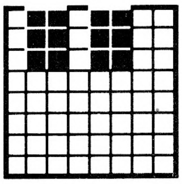
Seuraavilla sivuilla esitetyt CPC 464:n merkkivalikoiman merkit on esitetty siinä muodossa, jossa ne tulostuvat kuvaruutuun 8 x 8 merkkisolumatriisina. Käyttäjä voi määritellä lisäksi omia merkkejään, jotka muiden merkkien tapaan voidaan ryhmitellä isommiksi kokonaisuuksiksi. Aihetta käsitellään tarkemmin luvussa 8 SYMBOL-käskyn kohdalla.



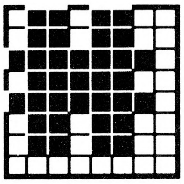
32 &H20
&X00100000



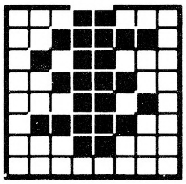
33
&H21
&X00100001



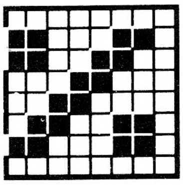
34
&H22
&X00100010



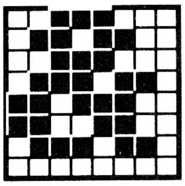
35
&H23
&X00100011



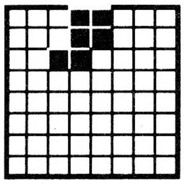
36
&H24
&X00100100



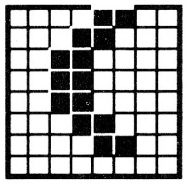
37
&H25
&X00100101



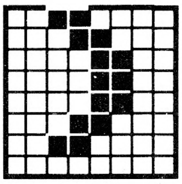
38
&H26
&X00100110



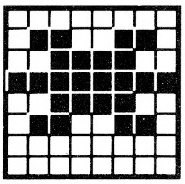
39
&H27
&X00100111



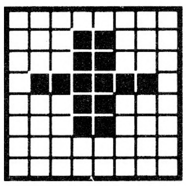
40
&H28
&X00101000



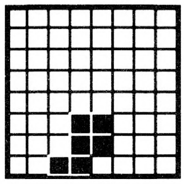
41
&H29
&X00101001



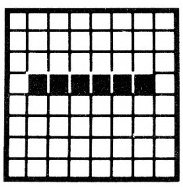
42
&H2A
&X00101010



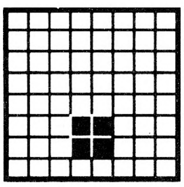
43
&H2B
&X00101011



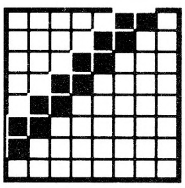
44
&H2C
&X00101100



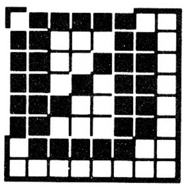
45
&H2D
&X00101101



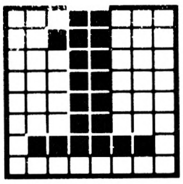
46
&H2E
&X00101110

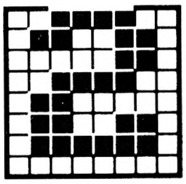


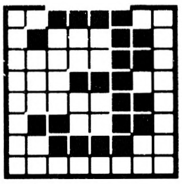
47
&H2F
&X00101111

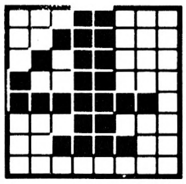


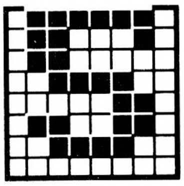
48
&H30
&X00110000

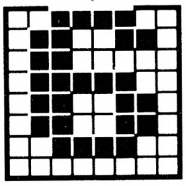

49
&H31
&X00110001

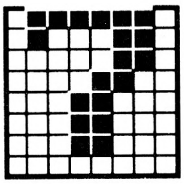

50
&H32
&X00110010

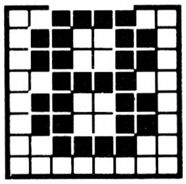

51
&H33
&X00110011

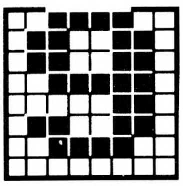

52
&H34
&X00110100

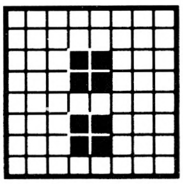

53
&H35
&X00110101

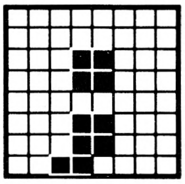

54
&H36
&X00110110

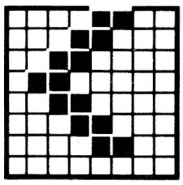

55
&H37
&X00110111

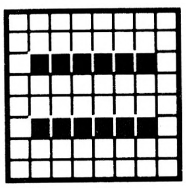

56
&H38
&X00111000

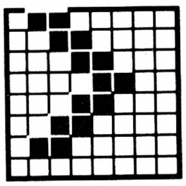

57
&H39
&X00111001

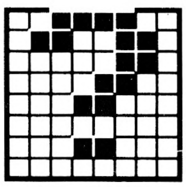

58
&H3A
&X00111010

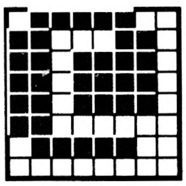

59
&H3B
&X00111011

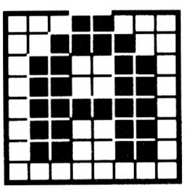

60
&H3C
&X00111100

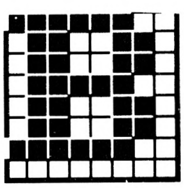

61
&H3D
&X00111101

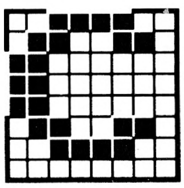

62
&H3E
&X00111110

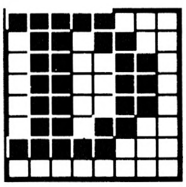

63
&H3F
&X00111111

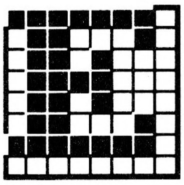

64
&H40
&X01000000


65
&H41
&X01000001

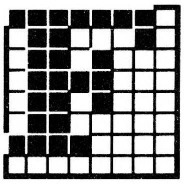

66
&H42
&X01000010


67
&H43
&X01000011

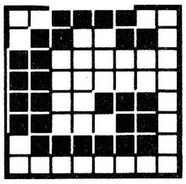

68
&H44
&X01000100



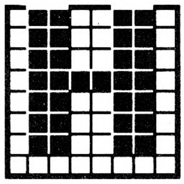
69
&H45
&X01000101



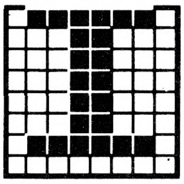
70
&H46
&X01000110



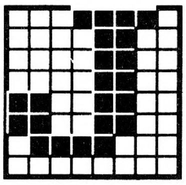
71
&H47
&X01000111



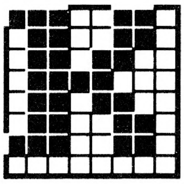
72
&H48
&X01001000



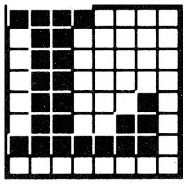
73
&H49
&X01001001



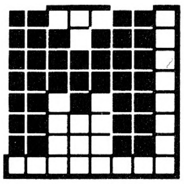
74
&H4A
&X01001010



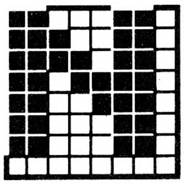
75
&H4B
&X01001011



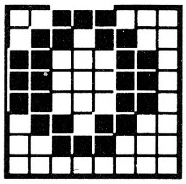
76
&H4C
&X01001100



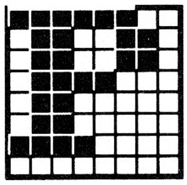
77
&H4D
&X01001101



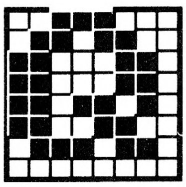
78
&H4E
&X01001110



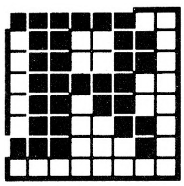
79
&H4F
&X01001111



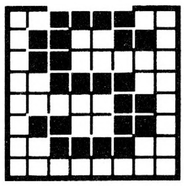
80
&H50
&X01010000



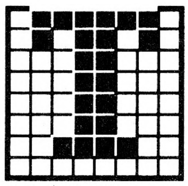
81
&H51
&X01010001



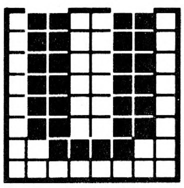
82
&H52
&X01010010



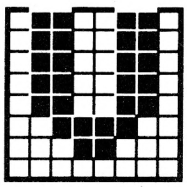
83
&H53
&X01010011



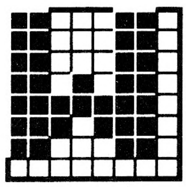
84
&H54
&X01010100



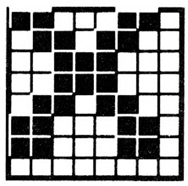
85
&H55
&X01010101



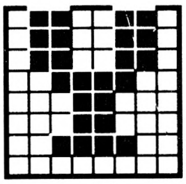
86
&H56
&X01010110

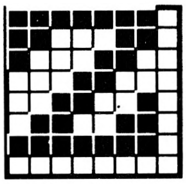


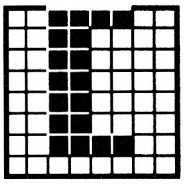
87
&H57
&X01010111

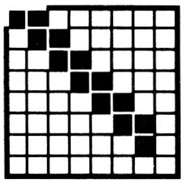


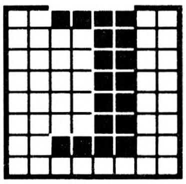
88
&H58
&X01011000

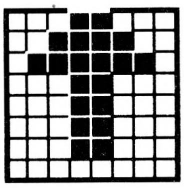

89
&H59
&X01011001

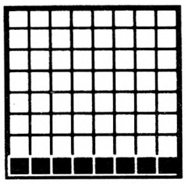

90
&H5A
&X01011010

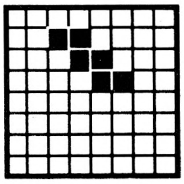

91
&H5B
&X01011011

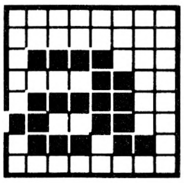

92
&H5C
&X01011100

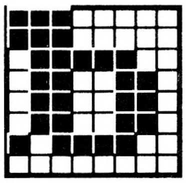

93
&H5D
&X01011101

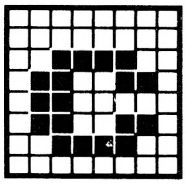

94
&H5E
&X01011110

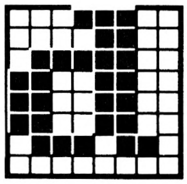

95
&H5F
&X01011111

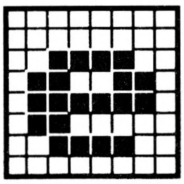

96
&H60
&X01100000

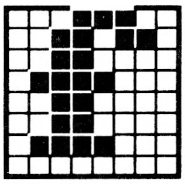

97
&H61
&X01100001

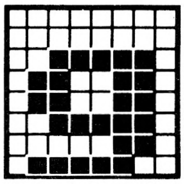

98
&H62
&X01100010

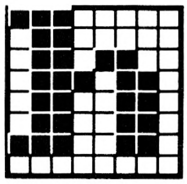

99
&H63
&X01100011

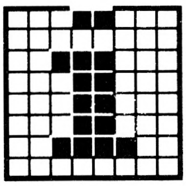

100
&H64
&X01100100

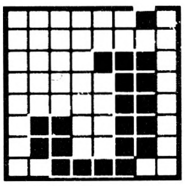

101
&H65
&X01100101

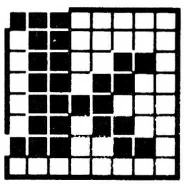

102
&H66
&X01100110

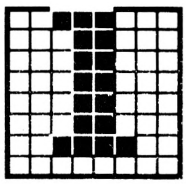

103
&H67
&X01100111

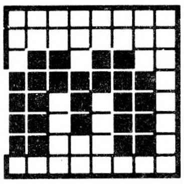

104
&H68
&X01101000


105
&H69
&X01101001

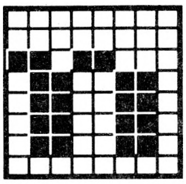

106
&H6A
&X01101010


107
&H6B
&X01101011

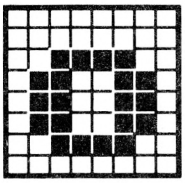

108
&H6C
&X01101100



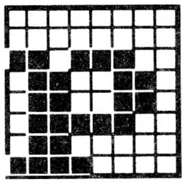
109
&H6D
&X01101101



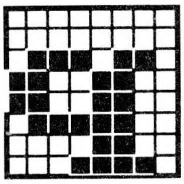
110
&H6E
&X01101110



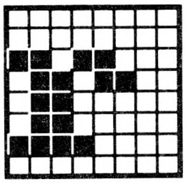
111
&H6F
&X01101111



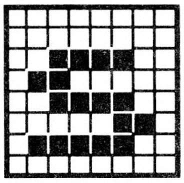
112
&H70
&X01110000



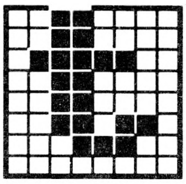
113
&H71
&X01110001



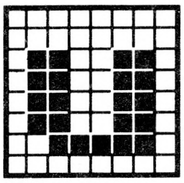
114
&H72
&X01110010



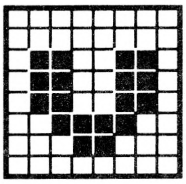
115
&H73
&X01110011



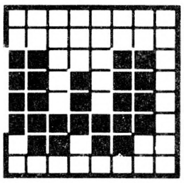
116
&H74
&X01110100



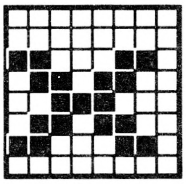
117
&H75
&X01110101



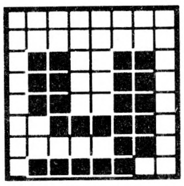
118
&H76
&X01110110



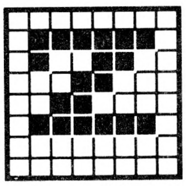
119
&H77
&X01110111



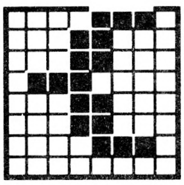
120
&H78
&X01111000



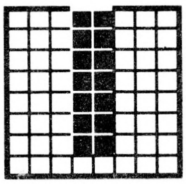
121
&H79
&X01111001



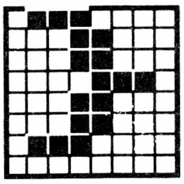
122
&H7A
&X01111010



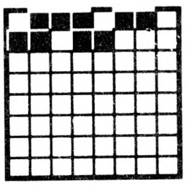
123
&H7B
&X01111011



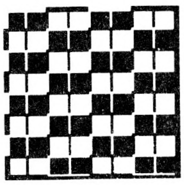
124
&H7C
&X01111100



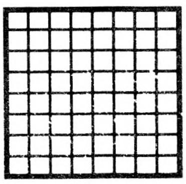
125
&H7D
&X01111101



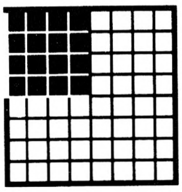
126
&H7E
&X01111110

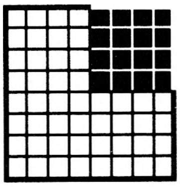


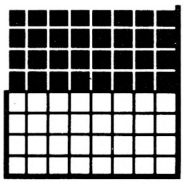
127
&H7F
&X01111111

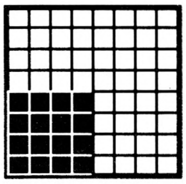


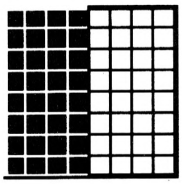
128
&H80
&X10000000

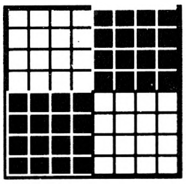

129
&H81
&X10000001

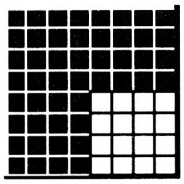

130
&H82
&X10000010

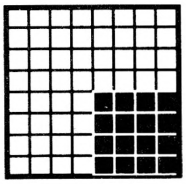

131
&H83
&X10000011

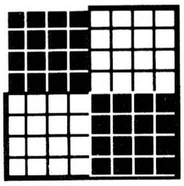

132
&H84
&X10000100

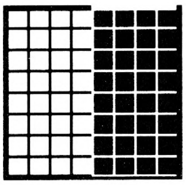

133
&H85
&X10000101

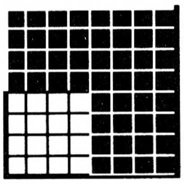

134
&H86
&X10000110

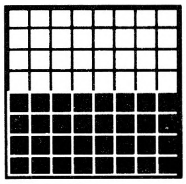

135
&H87
&X10000111

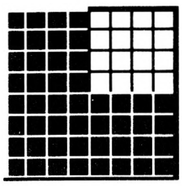

136
&H88
&X10001000

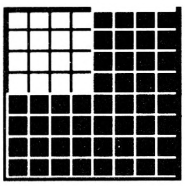

137
&H89
&X10001001

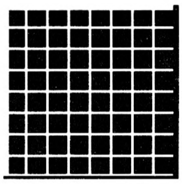

138
&H8A
&X10001010

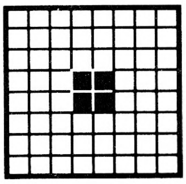

139
&H8B
&X10001011

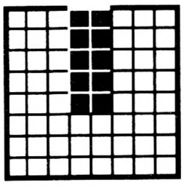

140
&H8C
&X10001100

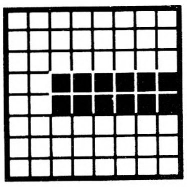

141
&H8D
&X10001101

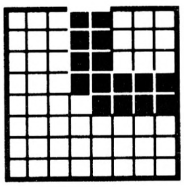

142
&H8E
&X10001110

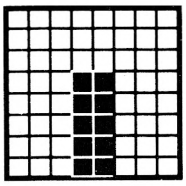

143
&H8F
&X10001111

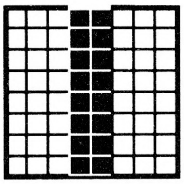

144
&H90
&X10010000

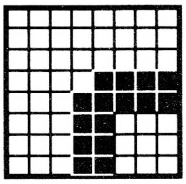

145
&H91
&X10010001

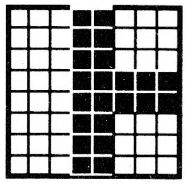

146
&H92
&X10010010

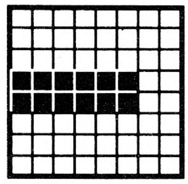

147
&H93
&X10010011

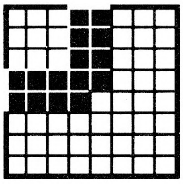

148
&H94
&X10010100

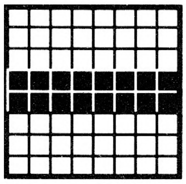

149
&H95
&X10010101

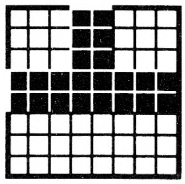

150
&H96
&X10010110

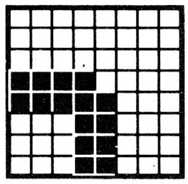

151
&H97
&X10010111

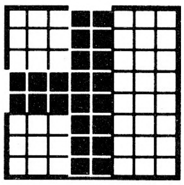

152
&H98
&X10011000

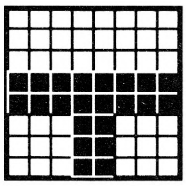

153
&H99
&X10011001

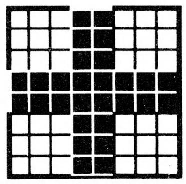

154
&H9A
&X10011010

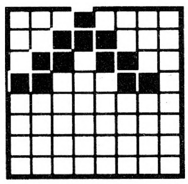

155
&H9B
&X10011011

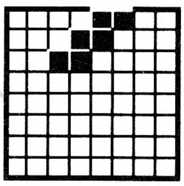

156
&H9C
&X10011100

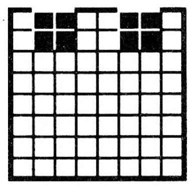

157
&H9D
&X10011101

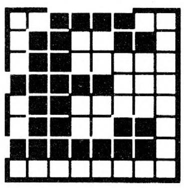

158
&H9E
&X10011110

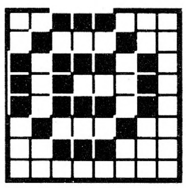

159
&H9F
&X10011111

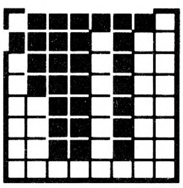

160
&HA0
&X10100000

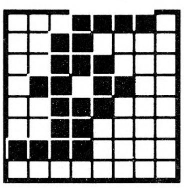

161
&HA1
&X10100001

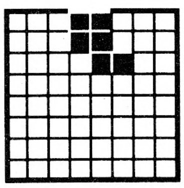

162
&HA2
&X10100010

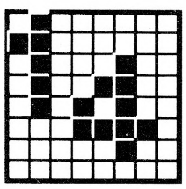

163
&HA3
&X10100011

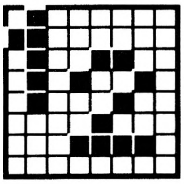

164
&HA4
&X10100100


165
&HA5
&X10100101

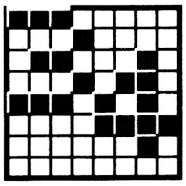

166
&HA6
&X10100110


167
&HA7
&X10100111

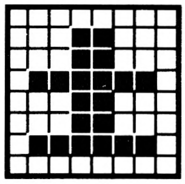

168
&HA8
&X10101000



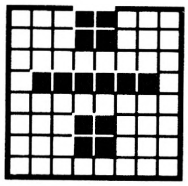
169
&HA9
&X10101001



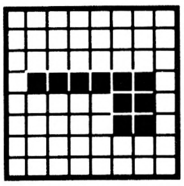
170
&HAA
&X10101010



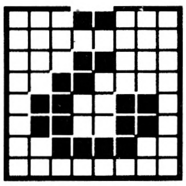
171
&HAB
&X10101011



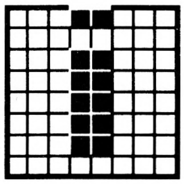
172
&HAC
&X10101100



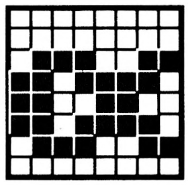
173
&HAD
&X10101101



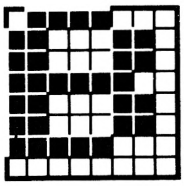
174
&HAE
&X10101110



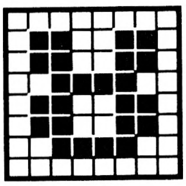
175
&HAF
&X10101111



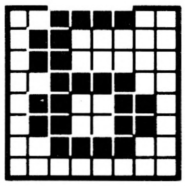
176
&HB0
&X10110000



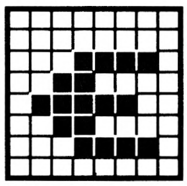
177
&HB1
&X10110001



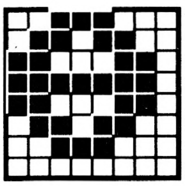
178
&HB2
&X10110010



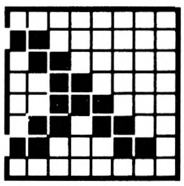
179
&HB3
&X10110011



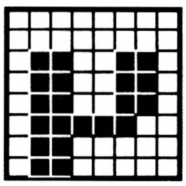
180
&HB4
&X10110100



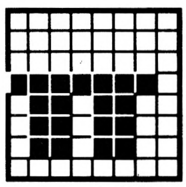
181
&HB5
&X10110101



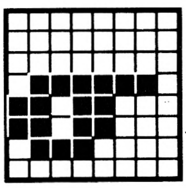
182
&HB6
&X10110110



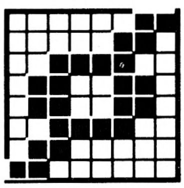
183
&HB7
&X10110111



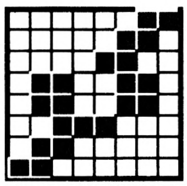
184
&HB8
&X10111000



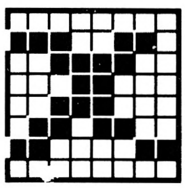
185
&HB9
&X10111001



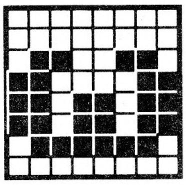
186
&HBA
&X10111010



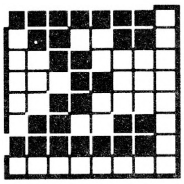
187
&HBB
&X10111011



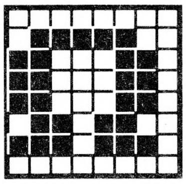
188
&HBC
&X10111100



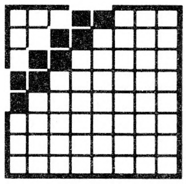
189
&HBD
&X10111101



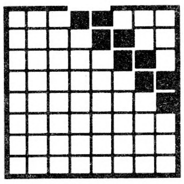
190
&HBE
&X10111110



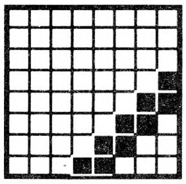
191
&HBF
&X10111111



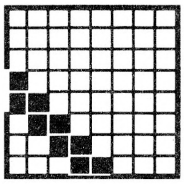
192
&HC0
&X11000000



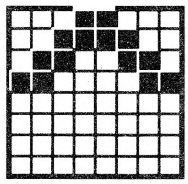
193
&HC1
&X11000001



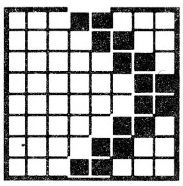
194
&HC2
&X11000010



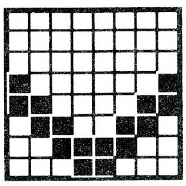
195
&HC3
&X11000011



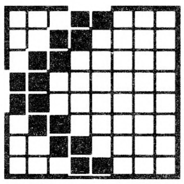
196
&HC4
&X11000100



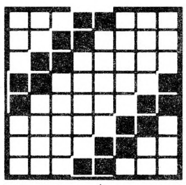
197
&HC5
&X11000101



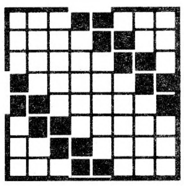
198
&HC6
&X11000110



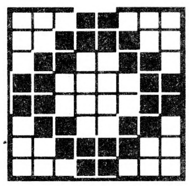
199
&HC7
&X11000111



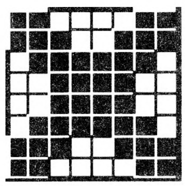
200
&HC8
&X11001000



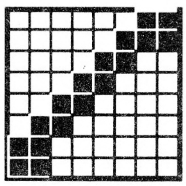
201
&HC9
&X11001001



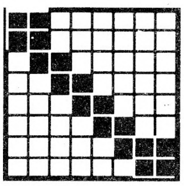
202
&HCA
&X11001010



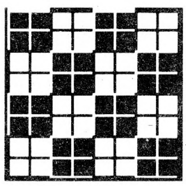
203
&HCB
&X11001011



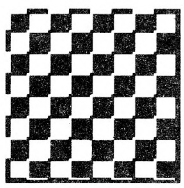
204
&HCC
&X11001100



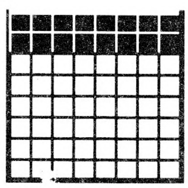
205
&HCD
&X11001101



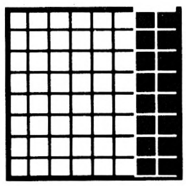
206
&HCE
&X11001110



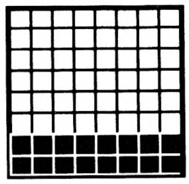
207
&HCF
&X11001111



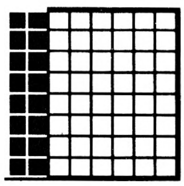
208
&HD0
&X11010000



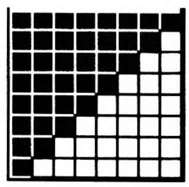
209
&HD1
&X11010001



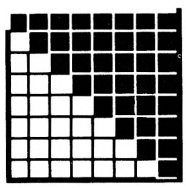
210
&HD2
&X11010010



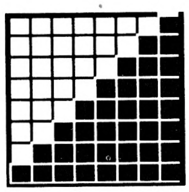
211
&HD3
&X11010011



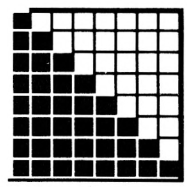
212
&HD4
&X11010100



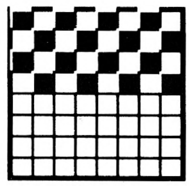
213
&HD5
&X11010101



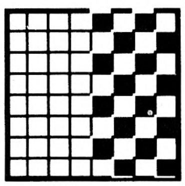
214
&HD6
&X11010110



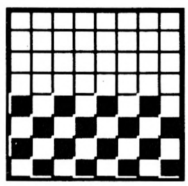
215
&HD7
&X11010111



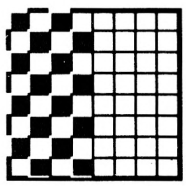
216
&HD8
&X11011000



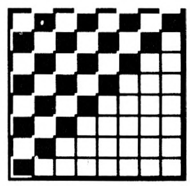
217
&HD9
&X11011001



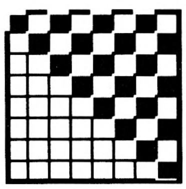
218
&HDA
&X11011010



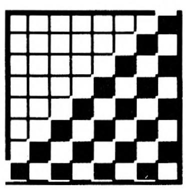
219
&HDB
&X11011011



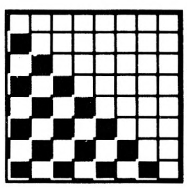
220
&HDC
&X11011100



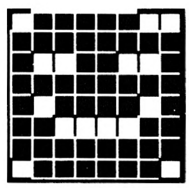
221
&HDD
&X11011101



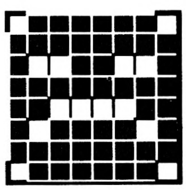
222
&HDE
&X11011110



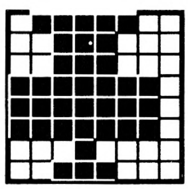
223
&HDF
&X11011111



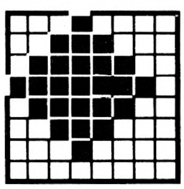
224
&HE0
&X11100000



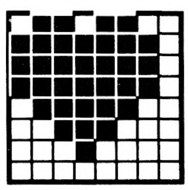
225
&HE1
&X11100001



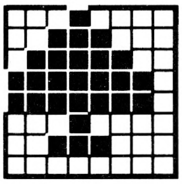
226
&HE2
&X11100010



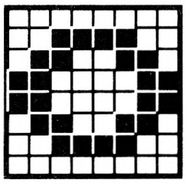
227
&HE3
&X11100011



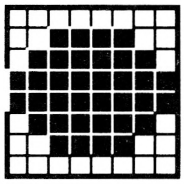
228
&HE4
&X11100100



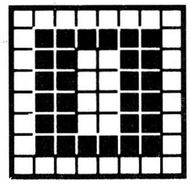
229
&HE5
&X11100101



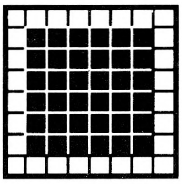
230
&HE6
&X11100110



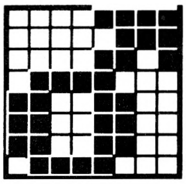
231
&HE7
&X11100111



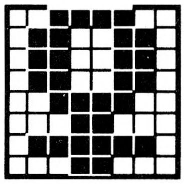
232
&HE8
&X11101000



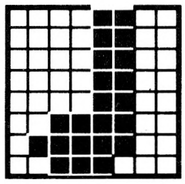
233
&HE9
&X11101001



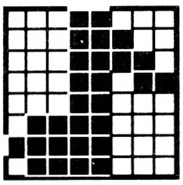
234
&HEA
&X11101010



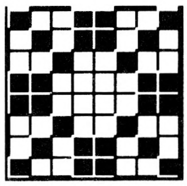
235
&HEB
&X11101011



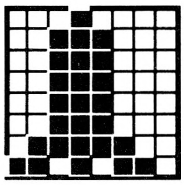
236
&HEC
&X11101100



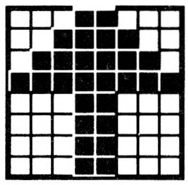
237
&HED
&X11101101



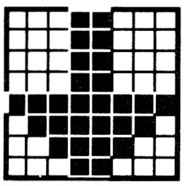
238
&HEE
&X11101110



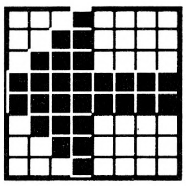
239
&HEF
&X11101111



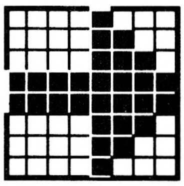
240
&HF0
&X11110000



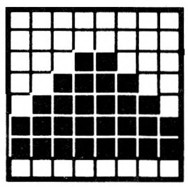
241
&HF1
&X11110001



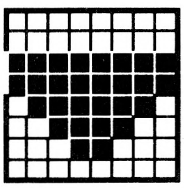
242
&HF2
&X11110010



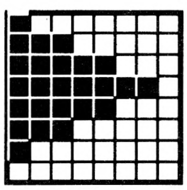
243
&HF3
&X11110011



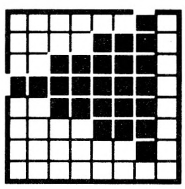
244
&HF4
&X11110100



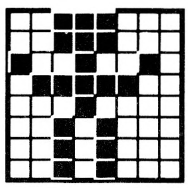
245
&HF5
&X11110101



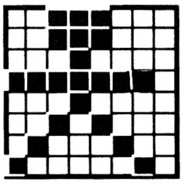
246
&HF6
&X11110110



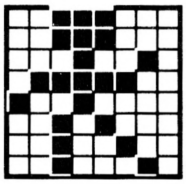
247
&HF7
&X11110111



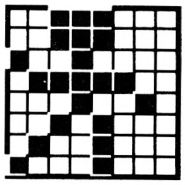
248
&HF8
&X11111000



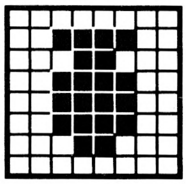
249
&HF9
&X11111001



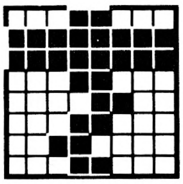
250
&HFA
&X11111010



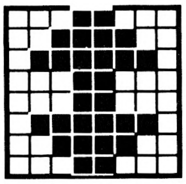
251
&HFB
&X11111011



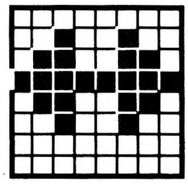
252
&HFC
&X11111100



253
&HFD
&X11111101

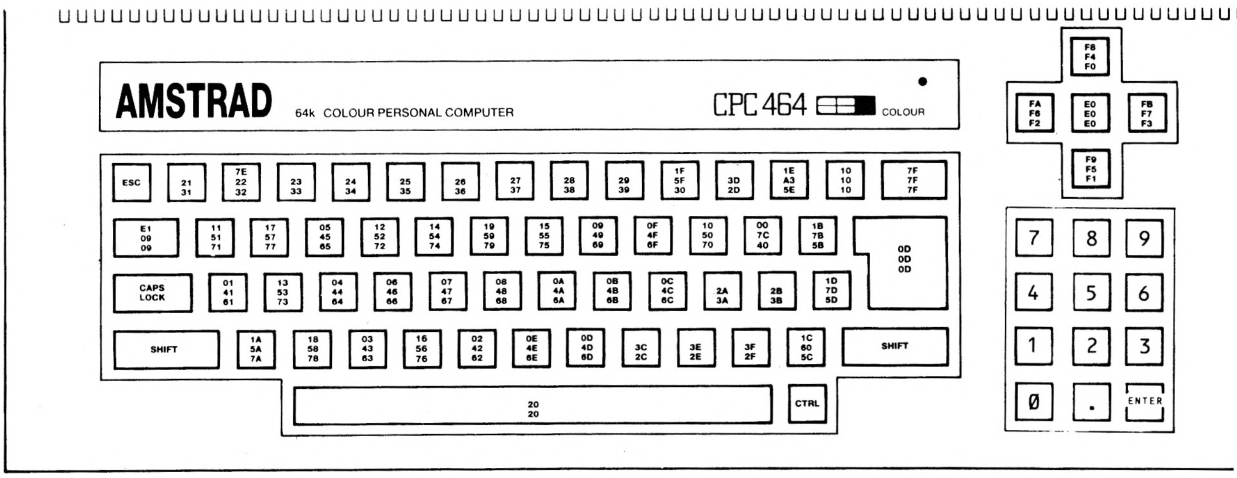


254
&HFE
&X11111110



255
&HFF
&X11111111

ASCII-OLETUSARVOT



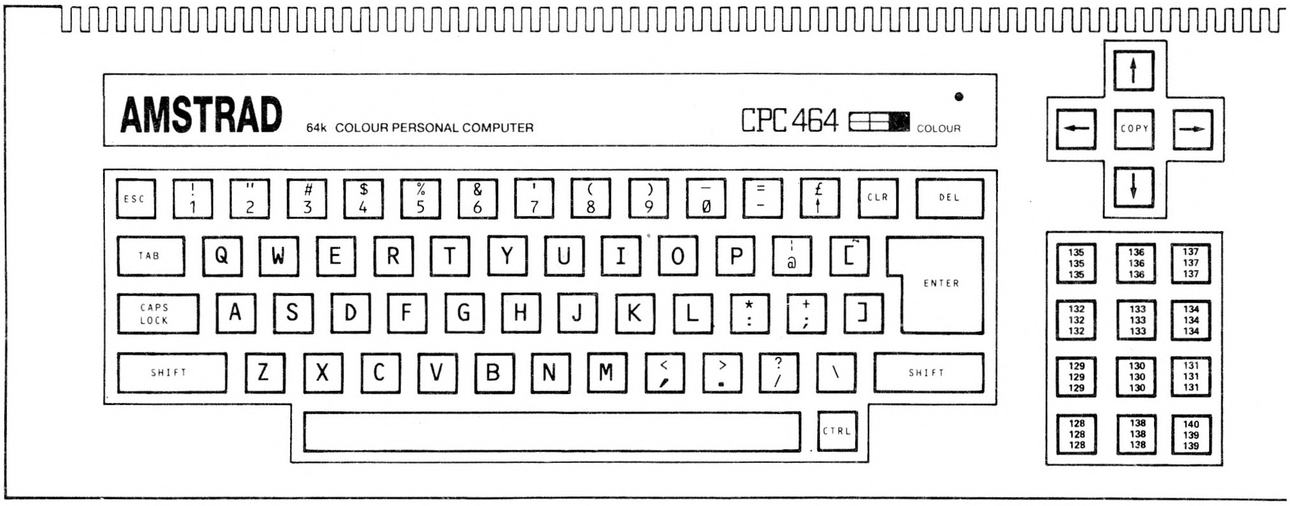
JOYSTICK 0

0B
0B
↑
FIRE 2 FIRE 1
08 ← 58 5A → 09
08 ← 58 5A → 09
↓
0A
0A

JOYSTICK 1

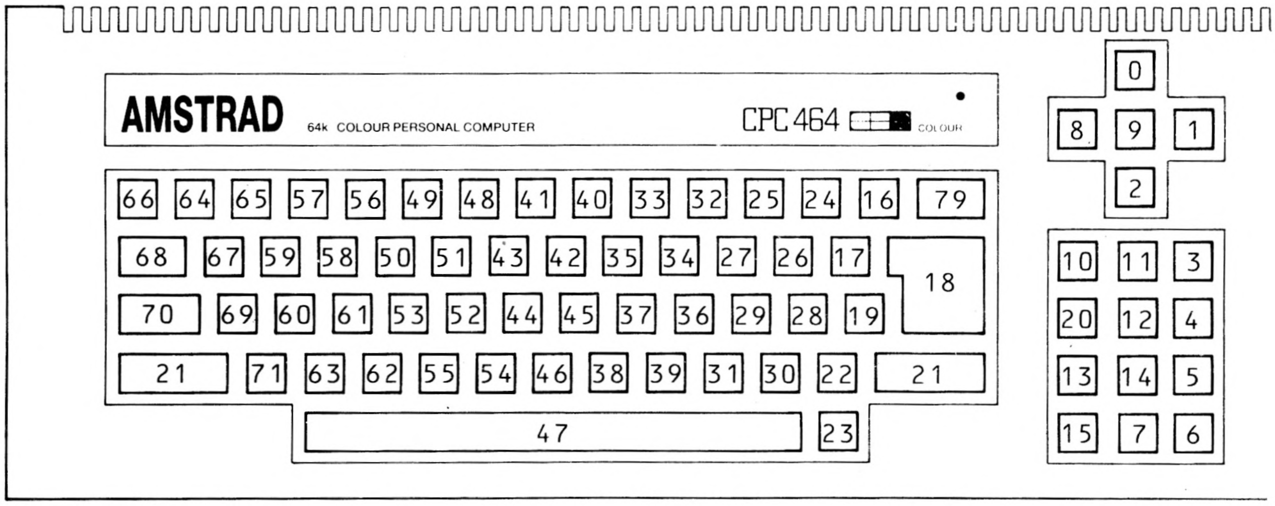
26
36
↑
FIRE 2 FIRE 1
12 07 06 14
52 ← 47 46 → 54
72 67 66 74
↓
25
35

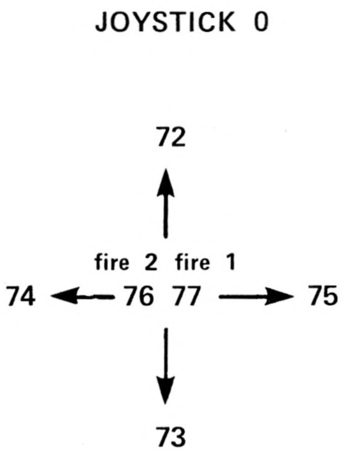
LAAJENNUSMERKIT, OLETUSPAIKAT JA -ARVOT

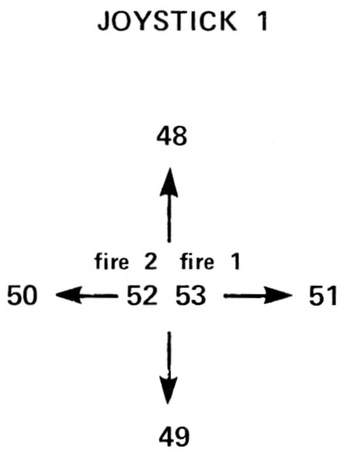


exp char	value	ascii
128	0	30
129	1	31
130	2	32
131	3	33
132	4	34
133	5	35
134	6	36
135	7	37
136	8	38
137	9	39
138	.	2E
139	[enter]	0D
140	run"	52,55,4E,22,0D

NÄPPÄINTEN JA SAUVAOHJAINTEN NÄPPÄINNUMEROT







KOKENEELLE KÄYTTÄJÄLLE TARKOITETTU KATSAUS KONEEN OMINAISUUKSISTA

CPC 464 on runsailla perustoiminnoilla ja laajennusmahdollisuuksilla varustettu edullinen väritietokone. Niinpä se sopii hyvin kokeneemmille käyttäjille sekä ensimmäiseksi tietokoneeksi.

CPC 464:N TÄRKEIMPIÄ OMINAISUUKSIA

Z80 CPU

Z80-mikroprosessori on maailman yleisin kotitietokoneissa käytettävä mikroprosessorityyppi. Siksi siihen löytyy runsaasti ohjelmistoa. Ohjelmatarjonta laajentuu moninkertaiseksi, koska CPC 464:ssä voidaan käyttää CP/M-käyttöjärjestelmän ohjelmia. CPC 464:ssä on ainutlaatuiset BASIC-ohjelmalla toteutettavat keskeytysmahdollisuudet AFTER- ja EVERY-käskyillä. Samoin siinä on muita "tosiaikaisia" ominaisuuksia äänten ja ajastinten ohjaamiseksi.

64K RAM

CPC 464:ssä on vakiona 64K RAM-muistia, josta yli 42K on käyttäjän käytettävissä. Tämä on saatu aikaan käyttämällä ROM-muistin kerrostustekniikkaa (overlay).

Näyttöruutu

CPC 464:n perustoimintoihin kuuluu kolme näyttömuotoa, joiden ominaisuuksia ovat mm. 80 merkkiä rivillä, 27 värin valikoima sekä 640 x 200 kuva-alkion erottelukyky.

Kirjoituskonenäppäimistö

"Oikea" kirjoituskonenäppäimistö, jonka lisäksi näppäimistössä on omat erilliset kohdistimen ohjausnäppäimensä sekä erillinen numeronäppäimistö. Tämä laajentaa toimintonäppäinten käyttömahdollisuutta entisestään. Skandinaaviset merkit ovat helposti ohjelmoitavissa.

Oma kasettinauhuri

CPC 464:n vakiovarusteisiin kuuluu oma kasettinauhuri. Tämä lisää käyttömukavuutta, koska ylimääräisiä liitäntöjä ei tarvitse tehdä ja tasoja ei tarvitse säätää. Tallennusnopeus on säädettävissä ohjelmakäskyllä joko 1 kilobaudiksi tai 2 kilobaudiksi. Ohjelmisto valitsee automaattisesti oikean lukunopeuden.

BASIC

CPC 464:n englantilainen BASIC kattaa laajan valikoiman vakiintuneita käskyjä ja funktioita, mutta sen lisäksi siinä on runsaasti lisätoimintoja äänitehosteiden ja grafiikkakuvioiden tekemiseksi. Lisäksi kiinteä laiteohjelmisto (firmware) tukee koneen BASICia varsin laajalti. Niinpä CPC 464:n BASIC onkin nopeampi ja monipuolisempi, kuin kotitietokoneen BASICilta osaisi odottaa.

Laaja merkkivalikoima

Näppäimistöltä saadaan suoraan käyttöön erittäin laaja 8-bittinen merkkivalikoima, johon sisältyy runsaasti symboleja ja grafiikkamerkkejä. Merkit saadaan käyttöön joko näppäimen painalluksella tai CHR$(n)-funktiolla.

Viiveajastimet

CPC 464:ssä on neljä viiveajastinta, joita käyttämällä voidaan muodostaa keskeytys- yms. toimintoja.

Käyttäjän määriteltävät näppäimet

Käyttäjä voi määritellä jopa 32 näppäintä, joille kullekin voidaan ohjelmoida enintään 32 merkkiä käsittävä merkkijono. Myös toistomahdollisuus (repeat) on ohjelmoitavissa näihin näppäimiin. Koneessa on täydellinen 255 merkkiä käsittävä merkkivalikoima (kaikki ASCII-merkit plus yli 100 muuta); valinnaisesti käyttäjän uudelleen määriteltävissä.

Alirutiinit

BASICilla voidaan kutsua useita assembler-alirutiineja.

Näyttömuodot

CPC 464:ssä on kolme näyttömuotoa (MODE):

a) Normaali näyttömuoto - MODE 1:
- 40 saraketta x 25 riviä; 4 "mustetta" (INK) tekstille
- 320 x 200 kuva-alkiota; kukin erikseen osoitettavissa neljällä eri värillä

b) Moniväriinen näyttömuoto - MODE 0:
- 20 saraketta x 25 riviä; 16 "mustetta" tekstille
- 160 x 200 kuva-alkiota; kukin erikseen osoitettavissa 16 eri värillä

c) Suuren erottelukyvyn näyttömuoto - MODE 2:
- 80 saraketta x 25 riviä; 2 "mustetta" tekstille
- 640 x 200 kuva-alkiota; kukin erikseen osoitettavissa kahdella eri värillä

Värien valinta

(Myös mustaa käsitellään värinä kaikkialla tässä käsikirjassa).

Reuna-alue (border) voidaan asettaa näyttömuodosta riippumatta minkä hyvänsä väriparin väriseksi. Se voi siis olla vilkkuva vaikkapa kahden värin välillä, tai se voi olla kiinteästi yhden värin värinen.

Käytettävissä olevien musteiden (INK) määrä riippuu valitusta näyttömuodosta. Kukin muste voidaan asettaa halutun väriparin väriseksi, eli se voi olla vilkkuva kahden värin välillä tai se voi olla kiinteästi tietyn värin värinen. Kullakin hetkellä käytettävissä olevien musteiden määrä riippuu siis valitusta näyttömuodosta. Tämän jälkeen tekstin paperi (PAPER), tekstin kynä (PEN) ja grafiikan kynä (PEN) voidaan asettaa käytettävissä olevan musteen väriseksi.

Kirjoittuva teksti voidaan asettaa joko läpinäkyväksi tai läpinäkymättömäksi eli se joko jättää huomioimatta paperin värin ja kirjoittuu grafiikan päälle, tai kirjoittuu kokonaan taustan päälle.

Ikkunat

Käyttäjä voi määritellä jopa 8 teksti-ikkunaa, johon merkit kirjoittuvat. Käyttäjä voi määritellä myös yhden grafiikkaikkunan, johon grafiikkamerkit tulostuvat. Ikkunat palautuvat oletusarvoihinsa näyttömuodon asetuskäskyllä (MODE).

> Huom. Jos teksti-ikkuna on koko kuvaruudun kokoinen (oletusarvo), niin kuvan vieritys tapahtuu nopeasti, koska vieristyksestä huolehtii laitteisto. Jos taas teksti-ikkuna on pienempi kuin koko kuvaruutu, niin vierittäminen on hitaampaa, koska tällöin vierittämisestä huolehtii ohjelmisto.

Kohdistin

Kohdistin estetään (disable) niiden jaksojen aikana, jolloin CPU ei odota tietojen syöttöä näppäimistöltä toimien täten automaattisena viestinä (prompt). Kohdistin näkyy kuvaruudussa käänteisenvärisenä pienenä ruutuna.

Moniääninen äänigeneraattori

CPC 464:n äänten muodostuksessa käytetään General Instrumentin AY8910-perheeseen kuuluvaa äänigeneraattoria. Siinä on 3 äänikanavaa, joista kunkin sävelkorkeus ja amplitudi on täysin itsenäisesti asetettavissa. Ääniin voidaan haluttaessa lisätä myös valkoista kohinaa. Äänikanavat saadaan stereoliitännästä vasempana kanavana, keskikanavana ja oikeana kanavana. Laitteen oma kaiutin toistaa eri kanavien äänet monofonisena sekoituksena.

Kullakin kolmella äänikanavalla muodostettujen äänten voimakkuus ja sävelkorkeus ovat säädettävissä ohjelmallisesti vaippakäskyillä. Äänigeneraattorin omia voimakkuusvaippoja ei yleensä käytetä.

Centronics-kirjoitinliitäntä

CPC 464:ssä on vakiomallinen Centronics-yhteensopiva kirjoittimen rinnakkaisliitäntä, jossa käytetään "busy"-signaalilinjaa kättelyoperaatioiden suorittamiseen.

Laajennettavuus

CPC 464:ään on saatavana useita laitteistoliitäntöjä erillis- tai kauttakulkutyyppisinä moduuleina ohjelmistolaajennuksia varten. Lisälaitteisiin kuuluvat mm. levyasemat, sarjaliitännät ROM-piirille sijoitettuine ohjausohjelmistoineen, jne.

Koordinaatit

Tekstin alkupiste (origin) on kuvaruudun vasemmassa yläkulmassa, ja kuvaruudun fyysiset merkkipaikat vaihtuvat valitun näyttömuodon mukaan.

Grafiikan alkupiste on vasemmassa alakulmassa, ja CPC 464 laskee grafiikkakohdistimen paikat ikään kuin näyttömuoto olisi aina MODE 2. Käytettävissä olevat musteet huomioidaan kuitenkin kulloinkin käytettävän todellisen näyttömuodon mukaan.

> Huom. Kullakin kuva-alkiolla on normaalissa näyttömuodossa (MODE 1) KAKSI vaakasuunnan osoitetta, joista kumpaakin voidaan käyttää. Monivärisessä näyttömuodossa (MODE 0) kullakin kuva-alkiolla on NELJÄ vaakasuunnan osoitetta, ja kuva-alkion paikan määrittämiseen voidaan käyttää mitä hyvänsä näistä neljästä osoitteesta.

Vaaka-akselin koordinaattiarvot ovat väliltä 0...399, jotka jaetaan kahdella ja katkaistaan (truncate), jotta tulokseksi saadaan fyysinen paikka väliltä 0...199. Tällä varmistetaan se, että kuvioiden halutut mittasuhteet säilyvät kuvaruudussa.

Laajennus-ROM-piirit

Kaikki ROM-piirit on sijoitettu muistin ylimpään 16 kilon lohkoon (jossa BASIC sijaitsee). CPC 464:n kiinteässä laiteohjelmistossa olevilla toiminnoilla on mahdollista kutsua jopa kahtasataaneljääkymmentä 16 kilon lisä ROM-piiriä (tällöin koneen perusversion lisäksi tarvitaan hiukan osoitteiden dekoodauslaitteistoa, joka on osa ROM-laajennuslaitteistoa).

YHTEENVETO

1 Laitteisto

1.1 CPC 464:n koteloon kuuluu:
Tietokone, näppäimistö, kasettinauhuri ja kaiutin. Lähtöliitännät RGB- ja luminanssisignaaleille.

1.1.1 LSI-piirit
- Z80A-prosessori; kellotaauus 4 MHz.
- 64 kilotavua (64 x 1 K) dynaamista RAM-muistia; virkistys kuvaruutumuistisaannein.
- 32 kilotavua ROM-muistia; sisältää BASICin ja käyttöjärjestelmän.
- Ajoitus-, värien muodostus ja DMA-piirit on sijoitettu erillisille logiikkapiireille.
- 6845 CRT-ohjain muodostaa pyyhkäisysignaalit video-RAM-piiriä varten.

Huom. Muistin karttarakenne on varsin monimutkainen, ja se muuttuu valitun näyttömuodon mukaan. CRT-ohjainta voidaan käyttää vaakasuuntaiseen vieritykseen (1/40 kuvaruudun leveydestä sivusuunnassa) ja pystyvieritykseen (8 pyyhkäisyjuovaa ylös ja alas), mikäli ohjelmisto edellyttää näiden vieritysten käyttöä. CRT-ohjaimeen kuuluvat parametrit valitsevat juovien määrän, kuvanopeuden, leveyden ja reuna-alueiden sijoituksen.

- Äänigeneraattori: General Instrumentin AY-3-8912. Äänigeneraattori on 3-kanavainen. Kustakin kanavasta otetaan yhtä voimakas signaali ja ne sekoitetaan monofoniseksi signaaliksi, joka kuuluu laitteen omasta kaiuttimesta. Äänenvoimakkuus on säädettävissä äänenvoimakkuuden säätimellä. Laitteen takana on myös stereoliitäntä, josta kanavat saadaan seuraavasti:
  - Vasen kanava = kanava A + 1/2 kanava C
  - Oikea kanava = kanava B + 1/2 kanava C

  Näppäimistön ja sauvaohjainportin tilatiedot välitetään myös äänigeneraattoripiirille.

- 8255 rinnakkais-I/O-piiri liittää väylän GI:n äänipiiriin. Se lukee myös näppäimistön, sauvaohjainportin ja lisälaiteliitäntöjen tilatiedot, ja ohjaa kasettinauhuria.

1.1.2 Ulkoiset liitännät

Korttiliitin yleislaajennuksiin (12) ja ulkoiselle kirjoittimelle (12) (Centronics-tyyppinen rinnakkaisliitäntä). Liitännät sauvaohjaimelle (14) (9-napainen D-tyyppinen), stereoäänen lähtöliitäntä (15), videolähtöliitäntä (10) (RGB ja tahdistus tai yhdistelmäsignaali tai luminanssi- ja tahdistussignaali).

1.2 Muut laitteiston laitteet

CPC 464-järjestelmässä voidaan käyttää kahdenlaisia videomonitoreja, joista kumpikin syöttää 5 V tietokoneelle. Lisävarusteena on saatavana virtalähdeyksikkö/UHF TV-modulaattori MP1.

Centronics-yhteensopivan kirjoittimen ja HiFi-laitteiston kytkemiseksi tarvitaan omat liitosjohtonsa.

1.3 Näyttötietojen erittely

Näytön ohjaukseen käytetään 16K muistia. Koneen värivalikoimaan kuuluu 27 väriä. Värivalikoimaan kuuluvien "musteiden" määrää riippuu valitusta näyttömuodosta. Useita musteita voidaan tarvittaessa asettaa samanväriseksi. Kuvaruudun kuva-alkiot määritellään tietyn musteen pisteiksi. (Huomaa, että tausta eli paperi tarvitsee yhden musteen käytettävissä olevista väreistä). Varsinaista kuva-aluetta ympäröi reuna-alue, jonka väri voidaan valita muista asetuksista riippumatta 27 värin valikoimasta.

Näyttö-muoto	**Musteiden määrä**	**Pisteitä pystysuunnassa**	**Pisteitä vaakasuunnassa**	**Merkkejä vaakasuunnassa**
MODE 1	4	200	320	40
MODE 2	2	200	640	80
MODE 0	16	200	160	20

Huom. Kuvat näkyvät mustavalkoisessa monitorissa harmaan eri sävyinä. Värien järjestys kirkkausjärjestyksessä alkaen tummimmasta on seuraava:

GREY LEVEL	COLOUR	GREY LEVEL	COLOUR
0	BLACK	13	WHITE
1	BLUE	14	PASTEL BLUE
2	BRIGHT BLUE	15	ORANGE
3	RED	16	PINK
4	MAGENTA	17	PASTEL MAGENTA
5	MAUVE	18	BRIGHT GREEN
6	BRIGHT RED	19	SEA GREEN
7	PURPLE	20	BRIGHT CYAN
8	BRIGHT MAGENTA	21	LIME GREEN
9	GREEN	22	PASTEL GREEN
10	CYAN	23	PASTEL CYAN
11	SKY BLUE	24	BRIGHT YELLOW
12	YELLOW	25	PASTEL YELLOW
		26	BRIGHT WHITE

(Katso myös sivuilla 26 ja 28 olevia taulukoita).

1.4 Muistikartta

CPC 464:n 64 kilon RAM-muisti on jaettu alla olevan muistikartan osoittamalla tavalla. Huomaa, että osa ROM-muistia on kerrostettu (overlay) kuvaruutua ohjaavan RAM-muistin kanssa. Näin käyttäjälle jää mahdollisimman suuri osa RAM-muistia BASIC-toimintoihin.

	0000H	
ROM SECTION 0 <		
	3FFFH	
	4000H	
	7FFFH	
	8000H	
	BFFFH	
	C000H	
ROM SECTION 1 <		> RAM [Screen 3]
	FFFFH	

Jos jossakin osoitteessa on sekä RAM- että ROM-muistia, niin LUKUTOIMINTO kohdistuu ROM-muistiin ja KIRJOITUSTOIMINTO kohdistuu RAM-muistiin. Kumpikin ROM-osa voidaan kytkeä pois toiminnasta, jolloin samassa osoitteessa olevaan RAM-muistiin voidaan kohdistaa lukutoiminto.

1.5 Laajennettavuus

1.5.1 Laajennus-ROM-piirit

CPC 464:ssä on mahdollisuus käyttää lisä-ROM-piirejä tietokoneen korteilla olevien ROM-piirien tilalla. Osoitteiden muodostukseen ja muistilohkojen valintaan käytettävät logiikkapiirit sijaitsevat laajennusväylään liitettävässä moduulissa, mutta kaikki tarvittavat signaalit tuodaan laajennusväylään.

1.5.2 Lisä-RAM-piirit

Tietokoneen korteilla olevien RAM-piirien paikalle voidaan kytkeä lisä-RAM-piirejä. Osoitteiden muodostukseen ja muistilohkojen valintaan käytettävät logiikkapiirit sijaitsevat laajennusväylään liitettävässä moduulissa, mutta kaikki tarvittavat signaalit tuodaan laajennusväylään. Näin muodostettu muisti on lukumuistia, ja se edellyttää syöttö/tulostusalueiden (I/O) erityisjärjestelyjä, jotta tietokoneesta voidaan kirjoittaa tähän lisä-RAM-muistiin.

1.5.3 I/O-lisäykset

Suurin osa I/O-porttien osoitteista on varattu tietokoneelle. Erityisesti osoitteen 7Fxx alapuolella olevia osoitteita ei pidä käyttää lainkaan. Seuraavia osoitteita voidaan käyttää ulkoisella laitteistolla:

F8xx, F9xx, FAxx, FBxx

Laajennusväylän oheislaitteiden on dekoodattava osoitteet linjoille A0 - A7 osoitteen A10 ollessa alhaalla. Laajennusväylän I/O-kanavat osoitealueella F800 - FBFF on varattu seuraavasti:

Osoitteet A0 - A7

00 - 7B	** Älä käytä **
7C - 7F	Varattu levyaseman liitännälle
80 - BB	** Älä käytä **
BC - BF	Varattu tuleviin käyttötarkoituksiin
C0 - DB	** Älä käytä **
DC - DF	Varattu tietoliikenneliitännöille
E0 - FF	Tarkoitettu käyttäjän oheislaitteille

Osoittamisessa on käytettävä Z80:n käskyjä, jotka sijoittavat B-rekisterin osoiteväylän ylemmälle puoliskolle (A15 - A8).

2. NÄPPÄIMISTÖ

Tietokone nollautuu täydellisesti painettaessa CTRL-, SHIFT- tai ESC-näppäimiä yhtäaikaa. Kaikkiin näppäimiin liittyy kiinteän laiteohjelmiston tukema toistotoiminto. Tämä koskee kaikkia niitä näppäimiä, jotka siirtävät kohdistinta tai tulostavat merkin kuvaruutuun. Poikkeuksena on erillisen numeronäppäimistön näppäimet.

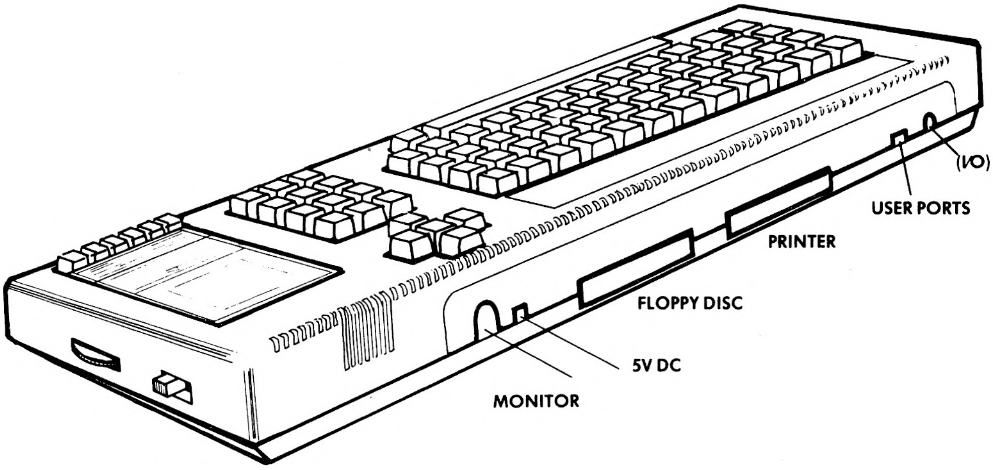
Ohjelma keskeytyy painettaessa ESC-näppäintä ja katkeaa painettaessa ESC-näppäintä toisen kerran peräkkäin. Keskeytettyä ohjelmaa voidaan jatkaa painamalla varsinaisen näppäimistön jotain näppäintä. CAPS LOCK-näppäimen toiminta on kahteen tilaan lukittuva. Samoin SHIFT LOCK lukittuu kahteen tilaan CTRL- ja CAPS LOCK-näppäimillä.

Kopiointikohdistin voidaan erottaa varsinaisesta tekstikohdistimesta painamalla SHIFT- ja kohdistimen ohjausnäppäimiä. Kopiointikohdistimen kohdalla oleva merkki saadaan käyttöön painamalla COPY-näppäintä.

Kohdistimen ohjausnäppäimet on tarkoitettu mm. syöttöpuskurin editoimista varten. Tämä voi käsittää useampia rivejä näytössä. Syöttökohta voidaan määritellä alkavaksi mistä hyvänsä kuvaruudun kohdasta käyttämällä kohdistimen ohjausnäppäimiä. Heti kun kone ottaa syöttötietoja näppäimistöltä, lukittuu kuvaruudun syöttökohta. Tällöin uusi teksti kirjoittuu vanhan kuvaruudussa olleen tekstin päälle.

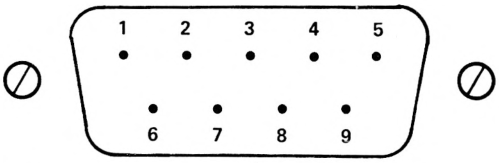
DEL-näppäin poistaa merkkejä kohdistimen vasemmalta puolelta ja CLR-näppäin kohdistimen kohdalta.

LIITE V: CPC 464:N TAKAPANEELISSA OLEVIEN LIITÄNTÖJEN NASTAJÄRJESTYS



PADDLE PORT CONNECTOR (9 PIN D)

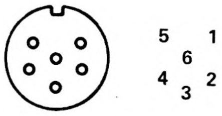
VIEWED FROM REAR



PIN 1	UP	PIN 6	FIRE 2
PIN 2	DOWN	PIN 7	FIRE 1
PIN 3	LEFT	PIN 8	COMMON
PIN 4	RIGHT	PIN 9	COM 2
PIN 5	SPARE		

VIDEO OUTPUT CONNECTOR (6 PIN DIN)

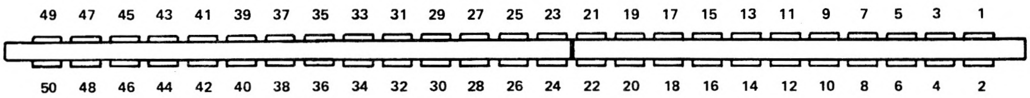
VIEWED FROM REAR



PIN 1	RED	PIN 4	SYNC
PIN 2	GREEN	PIN 5	GND
PIN 3	BLUE	PIN 6	LUM

EXPANSION PORT 50 WAY 0.1 EDGE CONNECTOR

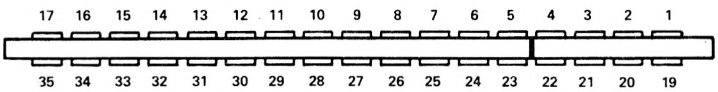
VIEWED FROM REAR



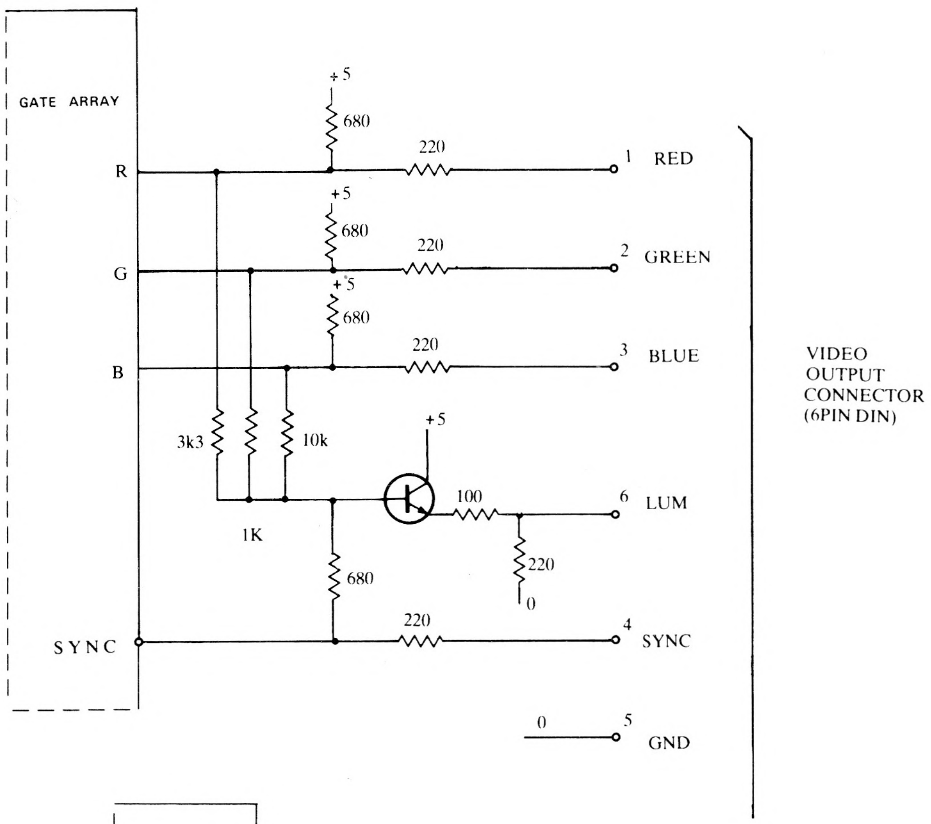
PIN 1	SOUND	PIN 18	A0	PIN 35	$\overline{INT}$
PIN 2	GND	PIN 19	D7	PIN 36	$\overline{NMI}$
PIN 3	A15	PIN 20	D6	PIN 37	$\overline{BUSRD}$
PIN 4	A14	PIN 21	D5	PIN 38	$\overline{BUSAK}$
PIN 5	A13	PIN 22	D4	PIN 39	$\overline{READY}$
PIN 6	A12	PIN 23	D3	PIN 40	$\overline{BUS\ RESET}$
PIN 7	A11	PIN 24	D2	PIN 41	$\overline{RESET}$
PIN 8	A10	PIN 25	D1	PIN 42	$\overline{ROMEN}$
PIN 9	A9	PIN 26	D0	PIN 43	ROMDIS
PIN 10	A8	PIN 27	+ 5v	PIN 44	$\overline{RAMRD}$
PIN 11	A7	PIN 28	$\overline{MREQ}$	PIN 45	RAMDIS
PIN 12	A6	PIN 29	$\overline{M1}$	PIN 46	CURSOR
PIN 13	A5	PIN 30	$\overline{RFSH}$	PIN 47	L. PEN
PIN 14	A4	PIN 31	$\overline{IORQ}$	PIN 48	$\overline{EXP}$
PIN 15	A3	PIN 32	$\overline{RD}$	PIN 49	GND
PIN 16	A2	PIN 33	$\overline{WR}$	PIN 50	ϕ
PIN 17	A1	PIN 34	$\overline{HALT}$		

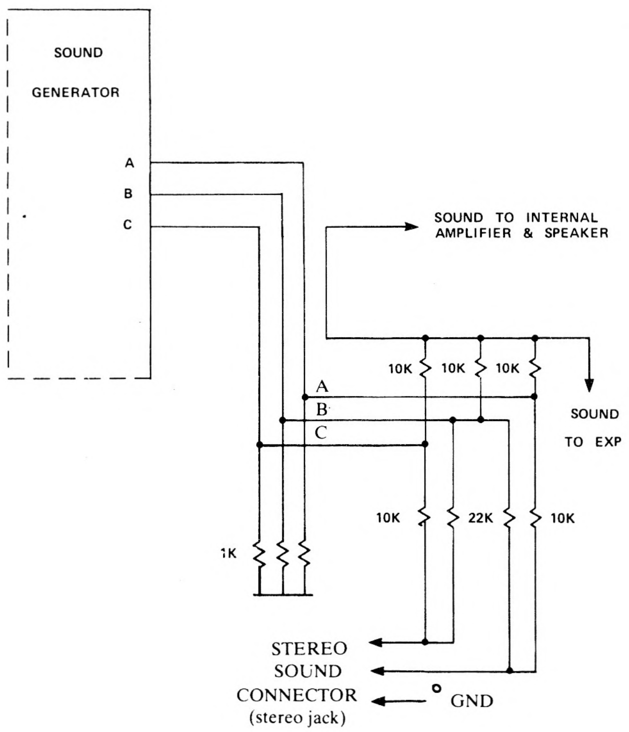
PRINTER PORT 34 WAY 0.1 EDGE CONNECTOR

VIEWED FROM REAR



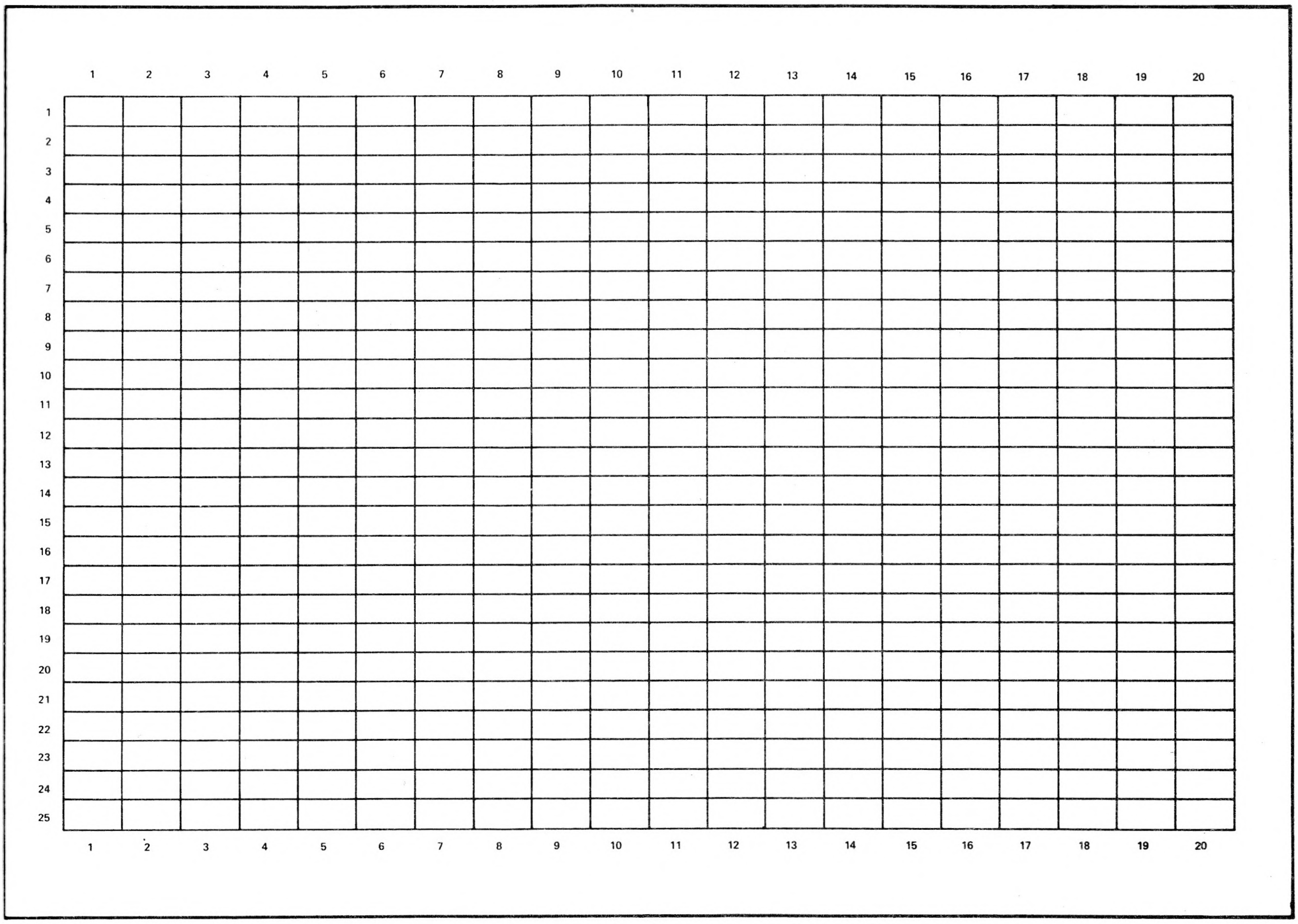
PIN 1	$\overline{STROBE}$	PIN 19	GND
PIN 2	D0	PIN 20	GND
PIN 3	D1	PIN 21	GND
PIN 4	D2	PIN 22	GND
PIN 5	D3	PIN 23	GND
PIN 6	D4	PIN 24	GND
PIN 7	D5	PIN 25	GND
PIN 8	D6	PIN 26	GND
PIN 9	D7	PIN 27	GND
PIN 11	BUSY	PIN 28	GND
PIN 14	GND	PIN 33	GND
PIN 16	GND	All other pins NC	


GATE ARRAY
R
G
B
SYNC
+5
680
220
1 RED
2 GREEN
3 BLUE
3k3
10k
1K
+5
100
6 LUM
220
0
680
220
4 SYNC
0
5 GND
VIDEO
OUTPUT
CONNECTOR
(6PIN DIN)

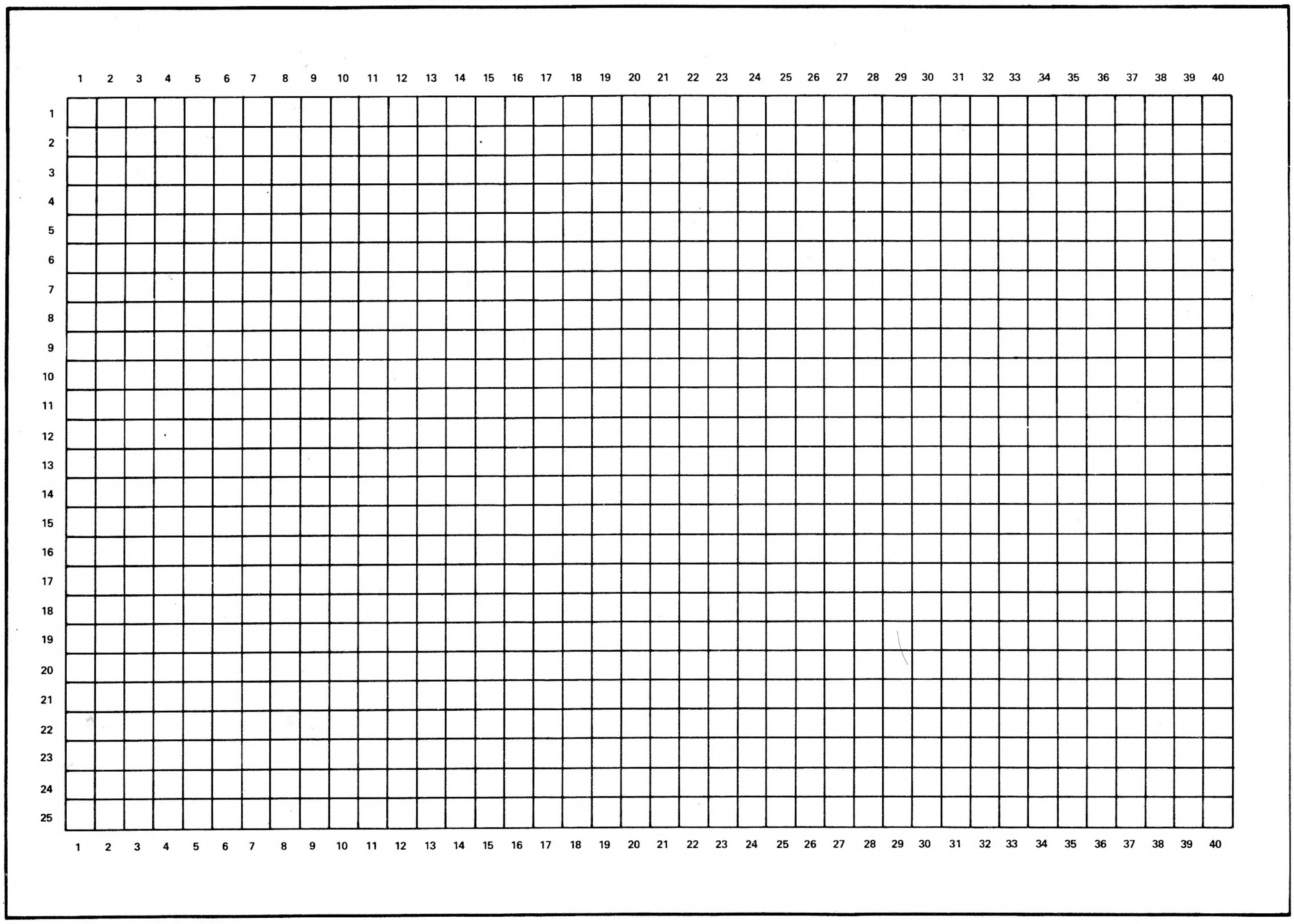

SOUND
GENERATOR
A
B
C
SOUND TO INTERNAL
AMPLIFIER & SPEAKER
10K
10K
10K
SOUND
TO EXP
A
B
C
10K
22K
10K
1K
STEREO
SOUND
CONNECTOR
(stereo jack)
GND

LIITE VI: RUUDUN RIVI- JA SARAKEMÄÄRÄT ERI NÄYTTÖMUODOISSA

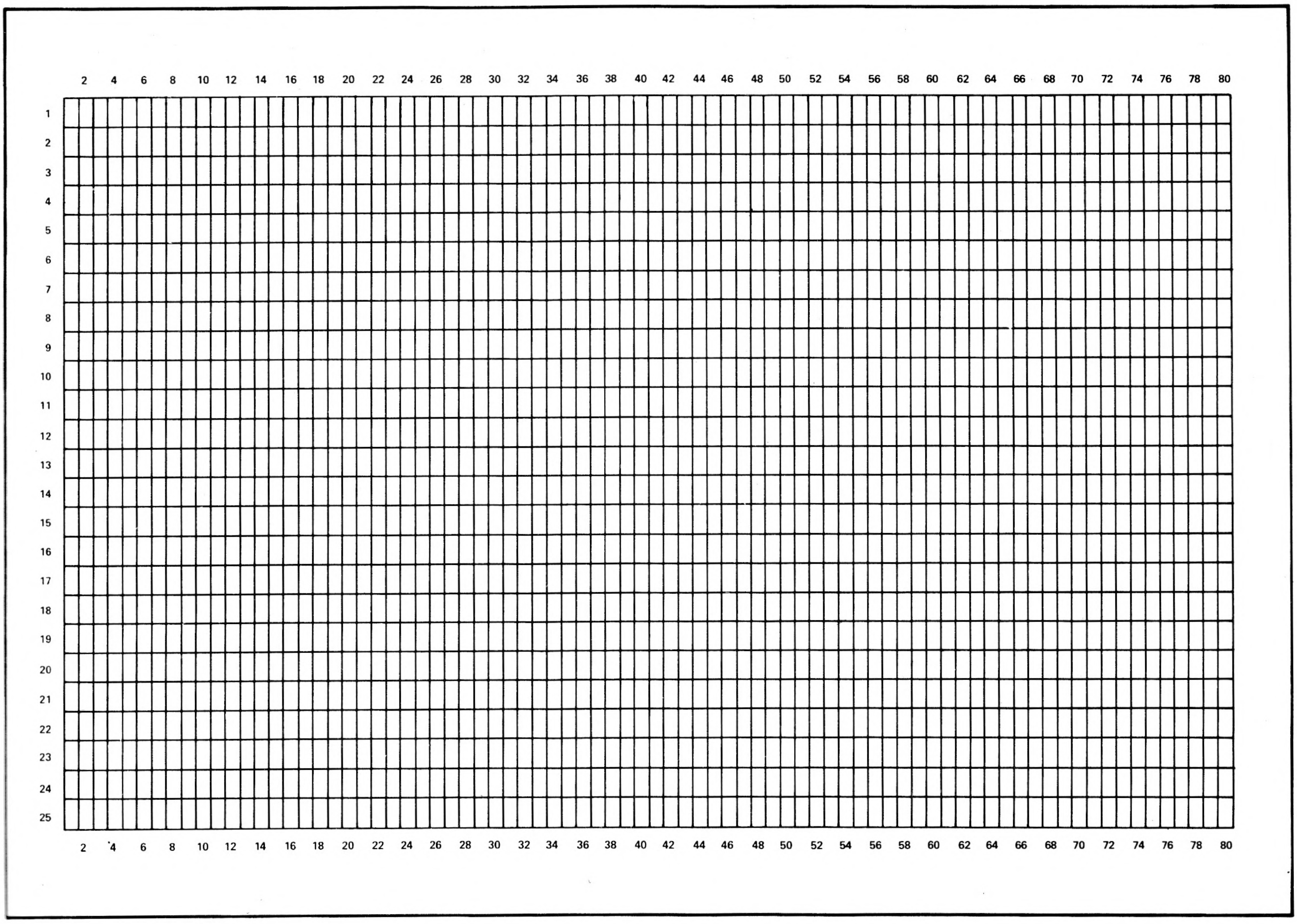
MODE 0 20 SARAKETTA



MODE 1 40 SARAKETTA



MODE 2 80 SARAKETTA



LIITE VII: NUOTIT JA SÄVELJAKSOT

Seuraavilla sivuilla olevissa taulukoissa on CPC 464:n säveljaksoarvoja (PERIOD) vastaavat nuotit (NOTE) tasajakoisella musiikkiasteikolla kahdeksan oktaavin alueella. CPC 464:n tuottama taajuus ei ole aivan tarkasti sama kuin vastaavan nuotin taajuus, koska SOUND-käskyssä annettavan säveljakson (TONE PERIOD) on oltava kokonaisluku. Taulukkoon on merkitty suhteellinen virhe (RELATIVE ERROR). Se on halutun ja todellisen taajuuden erotuksen sekä halutun taajuuden suhde eli (haluttu - todellinen)/(haluttu).

NOTE	FREQUENCY	PERIOD	RELATIVE ERROR	
C	32.703	3822	-0.007%	
C#	34.648	3608	+0.007%	
D	36.708	3405	-0.007%	
D#	38.891	3214	-0.004%	
E	41.203	3034	+0.009%	
F	43.654	2863	-0.016%	Octave -3
F#	46.249	2703	+0.009%	
G	48.999	2551	-0.002%	
G#	51.913	2408	+0.005%	
A	55.000	2273	+0.012%	
A#	58.270	2145	-0.008%	
B	61.735	2025	+0.011%	

NOTE	FREQUENCY	PERIOD	RELATIVE ERROR	
C	65.406	1911	-0.007%	
C#	69.296	1804	+0.007%	
D	73.416	1703	+0.022%	
D#	77.782	1607	-0.004%	
E	82.407	1517	+0.009%	
F	87.307	1432	+0.019%	Octave -2
F#	92.499	1351	-0.028%	
G	97.999	1276	+0.037%	
G#	103.826	1204	+0.005%	
A	110.000	1136	-0.032%	
A#	116.541	1073	+0.039%	
B	123.471	1012	-0.038%	

NOTE	FREQUENCY	PERIOD	RELATIVE ERROR	
C	130.813	956	+0.046%	
C#	138.591	902	+0.007%	
D	146.832	851	-0.037%	
D#	155.564	804	+0.058%	
E	164.814	758	-0.057%	
F	174.614	716	+0.019%	**Octave -1**
F#	184.997	676	+0.046%	
G	195.998	638	+0.037%	
G#	207.652	602	+0.005%	
A	220.000	568	-0.032%	
A#	233.082	536	-0.055%	
B	246.942	506	-0.038%	

NOTE	FREQUENCY	PERIOD	RELATIVE ERROR	
C	261.626	478	+0.046%	Middle C
C#	277.183	451	+0.007%	
D	293.665	426	+0.081%	
D#	311.127	402	+0.058%	
E	329.628	379	-0.057%	
F	349.228	358	+0.019%	**Octave 0**
F#	369.994	338	+0.046%	
G	391.995	319	+0.037%	
G#	415.305	301	+0.005%	
A	440.000	284	-0.032%	International A
A#	466.164	268	-0.055%	
B	493.883	253	-0.038%	

NOTE	FREQUENCY	PERIOD	RELATIVE ERROR	
C	523.251	239	+0.046%	
C#	554.365	225	-0.215%	
D	587.330	213	+0.081%	
D#	622.254	201	+0.058%	
E	659.255	190	+0.206%	
F	698.457	179	+0.019%	**Octave 1**
F#	739.989	169	+0.046%	
G	783.991	159	-0.277%	
G#	830.609	150	-0.328%	
A	880.000	142	-0.032%	
A#	932.328	134	-0.055%	
B	987.767	127	+0.356%	

NOTE	FREQUENCY	PERIOD	RELATIVE ERROR
C	1046.502	119	-0.374%
C#	1108.731	113	+0.229%
D	1174.659	106	-0.390%
D#	1244.508	100	-0.441%
E	1318.510	95	+0.206%
F	1396.913	89	-0.543%
F#	1479.978	84	-0.548%
G	1567.982	80	+0.350%
G#	1661.219	75	-0.328%
A	1760.000	71	-0.032%
A#	1864.655	67	-0.055%
B	1975.533	63	-0.435%

Octave 2

NOTE	FREQUENCY	PERIOD	RELATIVE ERROR
C	2093.004	60	+0.462%
C#	2217.461	56	-0.662%
D	2349.318	53	-0.390%
D#	2489.016	50	-0.441%
E	2637.021	47	-0.855%
F	2793.826	45	+0.574%
F#	2959.955	42	-0.548%
G	3135.963	40	+0.350%
G#	3322.438	38	+0.992%
A	3520.000	36	+1.357%
A#	3729.310	34	+1.417%
B	3951.066	32	+1.134%

Octave 3

NOTE	FREQUENCY	PERIOD	RELATIVE ERROR
C	4186.009	30	+0.462%
C#	4434.922	28	-0.662%
D	4698.636	27	+1.469%
D#	4978.032	25	-0.441%
E	5274.041	24	+1.246%
F	5587.652	22	-1.685%
F#	5919.911	21	-0.548%
G	6271.927	20	+0.350%
G#	6644.875	19	+0.992%
A	7040.000	18	+1.357%
A#	7458.621	17	+1.417%
B	7902.133	16	+1.134%

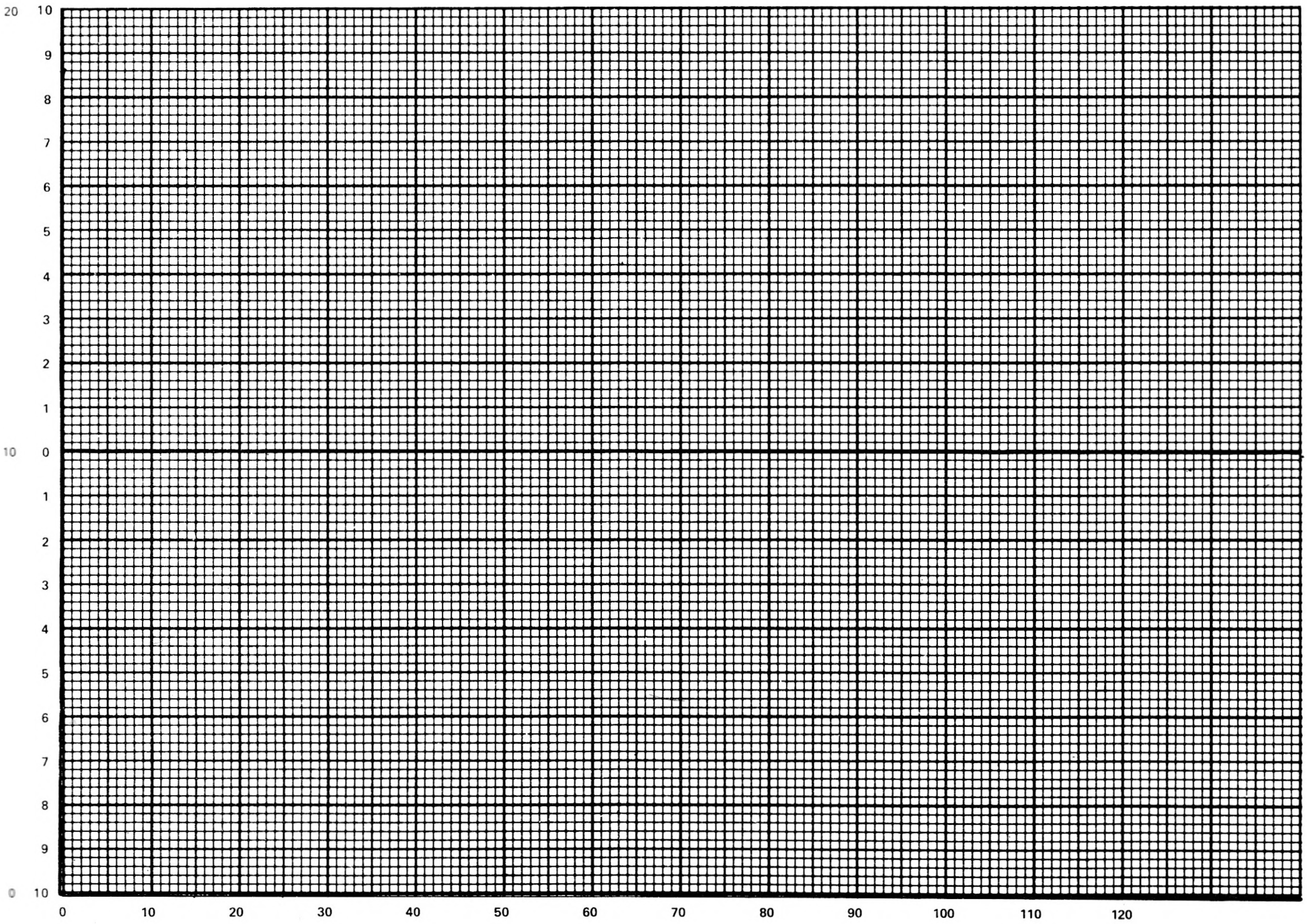
Octave 4

Arvot on laskettu kansainvälisestä A:sta (International A) seuraavasti:

```
FREQUENCY = 440*(2↑(OCTAVE+(N-10)/12))
   PERIOD = ROUND(125000/FREQUENCY)
```

jossa N on 1 C:lle, 2 C#:lle, 3 D:lle, jne.

APURUUDUKKO ÄÄNIVAIPPOJEN SUUNNITTELUA VARTEN



LIITE VIII: VIRHEKOODIT JA VARATUT SANAT

VIRHEILMOITUKSET JA VIRHEIDEN NUMEROT

Kun BASIC kohtaa ohjelmassa lauseen, sanan tai muuttujan jota se ei ymmärrä tai jota se ei pysty käsittelemään, se pysäyttää ohjelman suorituksen ja tulostaa kuvaruutuun virheilmoituksen. Virheilmoituksen muoto yleensä kertoo siitä, mikä on ongelman aiheuttaja. Joissakin tilanteissa BASIC ilmoittaa virheestä ja tulostaa virhellisen rivin kuvaruutuun, jos virhe on luonteeltaan ohjelman syöttämisen aikana tehty näppäilyvirhe. Täten virheen korjaaminen käy kätevästi ja nopeasti.

Yleisin kuvaruudussa näkyvä virheilmoitus on **Syntax error** (nro 2). Varsinkin huolimattomat näppäilijät näkevät tämän ilmoituksen varsin usein. Samalla BASIC ilmoittaa korjattavasta rivistä, mikäli se on tullut vastaan ohjelmaa suoritettaessa. Suorassa toimintatilassa BASIC ilmoittaa havaitsemastaan virheestä ja olettaa, että viimeksi näppäilty rivi näkyy kuvaruudussa, jolloin virhe on helposti korjattavissa.

Ohjelman suoritus voidaan ohjata halutulle riville, jos ohjelman alkuun laitetaan ON ERROR GOTO-käsky. Esimerkiksi alla olevassa tapauksessa tietokone ohjataan riville 1000, jos ohjelman suorituksen aikana tulee vastaan virhe:

```
10 ON ERROR GOTO 1000
...
...   (ohjelma)
...
1000 PRINT CHR$(7):MODE 2:INK 1,0:INK 0,9:CLS :LIST
```

Tällöin CPC 464 antaa virheen havaitessaan äänimerkin, tyhjentää kuvaruudun, vaihtaa näyttömuodoksi 80 sarakkeen näytön (MODE 2), asettaa sopivat värit ja tulostaa listauksen tutkittavaksi. Jos virhe on muotovirhe (Syntax error) se näkyy siten, että listauksen lopussa on korjattava rivi valmiina korjaustoimenpiteitä varten, mutta viestiä **Syntax error** ei näy kuvaruudussa.

Muista päättää ohjelma END-käskyllä viimeisellä rivillä ennen riviä 1000, jos haluat säilyttää tulokset kuvaruudussa.

BASIC ei muodosta virheilmoituksia, jos syöttötiedot ja näppäilyt ovat oikeita. Niinpä virheilmoituksen taustalla on aina jokin virhe, joka yleensä löytyy kuvaruutuun tulostuvan virheilmoituksen perusteella. Myös tässä yhteydessä pätee, että opit nopeasti virheistäsi. CPC 464 on sangen kärsivällinen opettaja, joten väsyt itse paljon ennemmin kuin CPC 464 menettää kärsivällisyytensä.

Seuraavassa luettelossa on kuvattu kaikki BASICin muodostamat virheilmoitukset virheiden numerojärjestyksessä. BASICin tuottaman virheilmoituksen jälkeen on lyhyt kuvaus ongelman mahdollisista aiheuttajista.

1 Unexpected NEXT (NEXT on väärässä kohdassa)
BASIC on löytänyt FOR...NEXT-silmukkaan kuulumattoman NEXT-käskyn, tai NEXT-käskyllä määritelty ohjausmuuttuja ei ole sama kuin FOR-käskyllä määritelty muuttuja.

2 Syntax error (Muotovirhe)
BASIC ei ymmärrä näppäiltyä riviä, koska sen rakenne ei ole sallittua muotoa.

3 Unexpected RETURN (RETURN on väärässä kohdassa)
BASIC on löytänyt aliohjelmaan kuulumattoman RETURN-käskyn.

4 DATA exhausted (DATA-arvot loppuneet)
BASIC on yrittänyt lukea READ-käskyllä DATA-käskyllä määriteltyjä DATA-arvoja pitemmälle kuin niitä on riittänyt (DATA-arvot ovat siis loppuneet kesken).

5 Improper argument (Epäkelpo argumentti)
Tällä virheilmoituksella on monta aiheuttajaa. Ilmoitus tarkoittaa, että funktion argumentin tai käskyn parametrin arvo on jollain tavalla epäkelpo.

6 Overflow (Ylitys)
Aritmeettisen operaation tulos on aiheuttanut ylityksen. Se voi olla liukulukuylitys, jolloin jostakin operaatiosta on tullut tulokseksi arvo, joka on suurempi kuin (noin) 1.7E-38. Ilmoitus voi aiheutua myös siitä, että yritys muuttaa liukuluku 16-bittiseksi etumerkillä varustetuksi kokonaisluvuksi on epäonnistunut.

7 Memory full (Muisti täynnä)
Tietokoneen muistissa parhaillaan oleva ohjelma tai sen muuttujat voivat olla liian suuria, tai ohjausrakenne on sijoitettu liian moneen kerrokseen sisäkkäin (sisäkkäin sijoitetut GOSUB-, WHILE- ja FOR-rakenteet).
Tämä virheilmoitus saadaan aikaan myös MEMORY-käskyllä, jos BASICin muistin yläraja yritetään asettaa liian alas tai epäsopivan korkeaksi. Avoimelle kasettitiedostolle on varattu puskuritilaa, ja tämä voi rajoittaa käytettävissä olevaa muistitilaa.

8 Line does not exist (Riviä ei ole)
BASIC ei löydä määriteltyä riviä.

9 Subscript out of range (Indeksi on alueen ulkopuolella)
Jokin taulukon mitoitukseen käytetyistä indekseistä on liian suuri tai liian pieni.

10 Array already dimensioned (Taulukko on jo määritelty)
Jokin DIM-lauseen taulukoista on jo aiemmin määritelty.

11 Division by zero (Jako nollalla)
Tämä voi esiintyä reaaliluvuilla tai kokonaisluvuilla tapahtuvan jakolaskun yhteydessä, kokonaiskertoimella suoritettavan laskun yhteydessä tai eksponenttilaskun yhteydessä.

12 Invalid direct command (Epäkelpo suora käsky)
Viimeksi näppäiltyä käskyä ei voi käyttää suorassa toimintatilassa.

13 Type mismatch (Tiedot ovat väärää tyyppiä)
Merkkijonon paikalla on yritetty käyttää numeroarvoa, tai päinvastoin, tai READ- tai INPUT-käskyn yhteydessä on tullut vastaan väärin muotoiltu luku.

14 String space full (Merkkijonoille varattu tila on täynnä)
Käyttäjä on luonut niin monta merkkijonoa, ettei tilaa ole enempää käytettävissä edes muistinsiivouksen (garbage collection) jälkeen.

15 String too long (Merkkijono on liian pitkä)
Merkkijonon pituus ylittää 255 merkkiä. Tällainen merkkijono voi muodostua, kun useampia merkkijonoja liitetään yhteen.

16 String expression too complex (Merkkijonolauseke on liian monimutkainen)
Merkkijonolausekkeet voivat muodostaa koko joukon väliarvoja. Kun näiden arvojen lukumäärä ylittää järkevän rajan, BASIC luovuttaa ja tulostaa tämän virheilmoituksen.

17 Cannot CONTinue (Ohjelmaa ei voi jatkaa CONT-käskyllä)
Muistissa parhaillaan olevaa ohjelmaa ei voida käynnistää uudelleen CONT-käskyllä syystä tai toisesta. CONT-käsky on tarkoitettu ohjelman käynnistämiseksi uudelleen STOP-käskyn jälkeen, kahdella ESC-näppäimen painalluksella suoritetun katkaisun jälkeen tai jonkin virheen jälkeen, eikä uudelleenkäynnistys ole mahdollista, jos tällä välillä ohjelmaan on tehty jotain muutoksia.

18 Unknown user function (Tuntematon käyttäjän funktio)
BASIC ei löydä DEF FN-käskyllä määriteltyä funktiota ohjelmassa vastaan tullutta FN-osaa vastaavasti.

19 RESUME missing (RESUME puuttuu)
Ohjelman loppu on tullut vastaan BASICin ollessa virheidenkäsittelytilassa eli ON ERROR GOTO-rutiinissa.

20 Unexpected RESUME (RESUME on väärässä kohdassa)
RESUME on sallittu vain virheidenkäsittelytilassa eli ON ERROR GOTO-rutiinin aikana.

21 Direct command found (On löytynyt suora käsky)
BASIC on löytänyt ohjelmaa kasetilta ladatessaan rivin, jossa ei ole rivinumeroa.

22 Operand missing (Operandi puuttuu)
BASIC on löytänyt epätäydellisen lausekkeen.

23 Line too long (Rivi on liian pitkä)
Rivi on osoittautunut liian pitkäksi muunnettaessa se BASICin sisäiseen muotoon.

24 EOF met (EOF on tullut vastaan)
On yritetty lukea kasetin syöttöreitin tiedoston lopun (End Of File) ohi.

25 File type error (Tiedostotyyppivirhe)
Luettu kasettitiedosto ei ole sopivaa tyyppiä. OPENIN-käskyä käytetään vain ASCII-tekstitiedostojen avaamiseen. LOAD-, RUN- yms. käskyjä käytetään vain käsiteltäessä SAVE-käskyllä luotuja tiedostotyyppejä.

26 NEXT missing (NEXT puuttuu)
BASIC ei löydä FOR-käskyyn sopivaa NEXT-käskyä.

27 File already open (Tiedosto on jo avattu)
BASIC on suorittanut OPENIN- tai OPENOUT-käskyn ennen kuin aiemmin avattu tiedosto on suljettu.

28 Unknown command (Tuntematon käsky)
BASIC ei tunne annettua käskyä.

29 WEND missing (WEND puuttuu)
BASIC ei löydä WHILE-käskyyn sopivaa WEND-käskyä.

30 Unexpected WEND (WEND on väärässä kohdassa)
BASIC on löytänyt WHILE-silmukkaan kuulumattoman WEND-käskyn tai senhetkiseen WHILE-silmukkaan sopimattoman WEND-käskyn.

BASIC-AVAINSANAT

Seuraavassa luettelossa on CPC 464:n BASIC-avainsanat. Ne ovat siis varattuja sanoja, eikä niitä voi käyttää muuttujien niminä.

ABS, AFTER, AND, ASC, ATN, AUTO

BIN$, BORDER

CALL, CAT, CHAIN, CHR$, CINT, CLEAR, CLG, CLOSEIN, CLOSEOUT, CLS, CONT, COS, CREAL

DATA, DEF, DEFINT, DEFREAL, DEFSTR, DEG, DELETE, DI, DIM, DRAW, DRAWR

EDIT, EI, ELSE, END, ENT, ENV, EOF, ERASE, ERL, ERR, ERROR, EVERY, EXP

FIX, FN, FOR, FRE

GOSUB, GOTO

HEX$, HIMEM

IF, INK, INKEY, INKEY$, INP, INPUT, INSTR, INT

JOY

KEY

LEFT$, LEN, LET, LINE, LIST, LOAD, LOCATE, LOG, LOG10, LOWER$

MAX, MEMORY, MERGE, MID$, MIN, MOD, MODE, MOVE, MOVER

NEXT, NEW, NOT

ON, ON BREAK GOSUB, ON BREAK STOP, ON ERROR GOTO, ON SQ, OPENIN, OPENOUT, OR, ORIGIN, OUT

PAPER, PEEK, PEN, PI, PLOT, PLOTR, POKE, POS, PRINT

RAD, RANDOMIZE, READ, RELEASE, REM, REMAIN, RENUM, RESTORE, RESUME, RETURN, RIGHT$, RND, ROUND, RUN

SAVE, SGN, SIN, SOUND, SPACE$, SPC, SPEED, SQ, SQR, STEP, STOP, STR$, STRING$, SWAP, SYMBOL

TAB, TAG, TAGOFF, TAN, TEST, TESTR, THEN, TIME, TO, TROFF, TRON

UNT, UPPER$, USING

VAL, VPOS

WAIT, WEND, WHILE, WIDTH, WINDOW, WRITE

XOR, XPOS

YPOS

ZONE

LIITE IX: OLETUSARVOT

Seuraavaan luetteloon on kerätty CPC 464:n oletusarvoja. Voit käyttää niitä hyväksesi esimerkiksi sellaisessa tilanteessa, kun haluat palauttaa tietokoneen toiminnan alkutilaan (oletusarvoihin) ohjelman ajon jälkeen. Tällä tavalla muistissa oleva ohjelma säilyy ennallaan. Olet esimerkiksi voinut seota väreissä niin, ettei kuvaruudussa näy mitään tekstejä. Tällöin saat tekstit näkyviin näppäilemällä PEN-, PAPER-, BORDER- ja INK-käskyjen oletusarvot.

- MODE 1
- INK 1,24 (teksti)
- INK 0,1 (paperi)
- PEN 1
- PAPER 0
- BORDER 1
- AUTO 10,10 (automaattinen rivinumerointi)
- RAD (mittayksikkö radiaaneja COS-, SIN- ja TAN-funktioiden yhteydessä)
- DIM (10)
- STEP 1
- VALMISVAIPAN (HARDWARE ENVELOPE) PORTAAN KESTO 2 sek.
- WINDOW #0 ja kohdistimen sijainti 1,1
- TIEDOSTON NIMEN (FILE NAME) maks. pituus 16 merkkiä
- ZONE 13
- SPC (0);
- RENUM 10,,10
- SPEED KEY 30,2 (10,10?)
- SPEED WRITE 0 (1000 baudia)
- SYMBOL AFTER 240 (merkit 240 - 255)
- TAG #0
- AFTER n,0

HAKEMISTO

ABS................................136
AFTER.........................136, 193
Alkupiste ks. ORIGIN
AND (JA)...........................105
Aritmetiikkaa.......................21
ASC................................136
ASCII...................51, 230, 242
Assembler..........................192
ATN................................137
AUTO.........................102, 137
Avainsanat...............14, 134, 268

BASIC............14, 76, 134, 197, 268
BASIC-tulkki.......................197
BIN$...............................137
BORDER...............27, 97, 137, 250

CALL...............................138
CAPS LOCK-näppäin...................11
CAT...........................72, 138
CHAIN, CHAIN MERGE.................139
CHR$.....................32, 51, 139
CINT...............................139
CLEAR..............................140
CLG................................140
CLOSEIN............................140
CLOSEOUT...........................140
CLR-näppäin.........................11
CLS...........................14, 141
CONT..........................90, 141
COPY-näppäin...................19, 62
COS................................141
CREAL..............................141
CTRL-näppäin..............11, 51, 188

DATA.........................101, 142
DEF FN.............................142
DEFINT, DEFREAL, DEFSTR............142
DEG................................143
DELETE.............................143
DEL-näppäin.........................10
DI (Disable Interrupt).............143
DIM...........................99, 144
DRAW..........................35, 144
DRAWR..............................144

EDIT......................18, 62, 145
Ehdolliset lauseet............87, 105
EI (Enable Interrupt)..............145
EI (NOT)...........................105
ELSE...............................152
END...........................20, 145

ENT......................43, 127, 145
ENTER-näppäin...................10, 53
ENV......................41, 126, 147
EOF................................148
ERASE..............................148
ERL, ERR...........................149
Erottimet
(rajamerkit)..........52, 77, 84, 135
ERROR..............................149
ESC-näppäin.........................12
Esittelykasetti...................5, 67
Eteenpäinkelauspainike (F.F.).......65
EVERY........................149, 194
EXP................................150

FIX................................150
FOR...........................21, 150
FRE...........................79, 150

GOSUB.........................38, 151
GOTO..........................15, 151
Grafiikka................32, 108, 133
Grafiikkakohdistin.................115

HEX$...............................151
HIMEM..............................151
Hitaampi tallennusnopeus............70

IF........................19, 97, 152
Ikkunat ks. WINDOW
Ikkunoiden suunnitteluruudukot 257-259
INK...........................26, 152
INKEY..............................153
INKEY$.............................153
INP................................154
INPUT.........................16, 154
INSTR..............................155
INT................................155
I/O (syöttö/tulostus)....40, 252, 254

Jakolasku...........................22
JOY................................156

Kaksi tiedonsiirtonopeutta..........70
Kasetin äänityksen estävät
kielekkeet..........................66
Kasetti...................5, 64, 75
Kasettien lataaminen........7, 67, 158
Kertolasku..........................22
Keskeytykset..................192, 193
KEY...........................58, 156
KEY DEF............................156

Kirjoitin..........132, 133, 248, 255
Kohdistin.........2, 57, 115, 188, 247
Kohina......................44, 126
Koneen käyttöjärjestelmä..........191
Koordinaatit..............32, 35, 248
Korjailu.......................18, 59
Korjailu kopiointikohdistimella 19, 60
Kuutiojuuri........................23
Kuvan kirkkaussäädin
(BRIGHTNESS)..................2, 3, 47
Käyttäjän määriteltävät näppäimet..246
Käyttäjän portti
(USER PORT)...............5, 130, 254

Laajennusmerkit...................243
Laajennus-ROM-piirit..........198, 249
Laitteisto.........................249
LEFT$..............................157
LEN................................157
LET................................157
Levyasema...............198, 248, 255
Liitännät..........................254
LINE INPUT.........................158
LIST....................15, 57, 158
Lisä-ROM-piirit....................252
LOAD.........................7, 158
LOCATE..................32, 96, 159
LOG................................159
LOG10..............................159
Loogiset lausekkeet...........87, 105
LOWER$.............................159
Lukuvirheet.........................73
Läpinäkyvä kirjoitustapa...........109

MAX................................160
MEMORY........................151, 160
MERGE..............................160
Merkintätapa.......................134
Merkit.........................53, 230
Merkkijonomuuttujat.........16, 17, 90
MID$...............................161
MIN................................161
MOD.................................86
MODE...............25, 110, 161, 246
Modulaattori/virtalähde (MP1)....3, 49
Moniääninen äänigeneraattori..121, 248
MOVE...............................161
MOVER..............................162
Muistikartta.......................251
Muotovirhe ks. Syntax error
Muuttujat.............16, 17, 84, 90

Neliöjuuri..........................22
NEW............................36, 162
NEXT...........................21, 162

Nollaus (palautus alkutilaan)....7, 46
Nuotit ja säveljaksot..............260
Näppäimistö...........10, 52, 242, 253
Näyttömuodot ks. MODE

Ohjausmerkit.......................188
Oletusarvot........................269
ON BREAK GOSUB.....................163
ON BREAK STOP......................163
ON ERROR GOTO..................164, 264
ON GOSUB, ON GOTO..................162
ON SQ GOSUB....................129, 164
OPENIN.............................164
OPENOUT............................165
Operaattorit..........21 - 25, 86 - 87
OR (TAI)...........................105
ORIGIN..........................38, 165
OUT................................165

PAPER......................26, 27, 166
PEEK...............................166
PEN........................26, 27, 166
PI.................................167
PLOT............................35, 167
PLOTR..............................168
POKE...............................168
POS................................169
Potenssiin korotus.............23, 24
PRINT............14, 79, 169, 170, 171
PRINT SPC......................103, 170
PRINT TAB.......................82, 170
PRINT USING................82, 83, 171
Pysäytys/kasetinpoistopainike
(STOP/EJECT)........................66
Pystypidon säädin (V-HOLD).......2, 47

RAD................................172
Rajamerkit
(erotusmerkit).........52, 77, 84, 135
RAM............................251, 252
RANDOMIZE..........................172
READ......................101, 142, 172
RELEASE...................124, 129, 173
REM............................112, 173
REMAIN.........................173, 195
RENUM......................93, 112, 174
RESTORE............................174
RESUME.............................174
RETURN..........................38, 175
Reuna-alue ks. BORDER
RIGHT$.............................175
RND................................175
ROUND..............................176
RUN.............................14, 176

Sanasto..............................202
Sauvaohjaimet
(JOYSTICK)...........5, 130, 242, 244
SAVE...................9, 71, 72, 177
SGN..................................177
SHIFT-näppäin.........................10
SIN..................................177
Skandinaaviset merkit...............108
SOUND..........40, 119, 178, 199, 260
SPACE$...............................178
SPC.............................103, 170
SPEED INK........................98, 178
SPEED KEY............................179
SPEED WRITE......................70, 179
SQ..............................128, 179
SQR..................................180
STEP.............................21, 150
Stereo......................40, 122, 199
STOP.................................180
STR$.................................180
STRING$..............................181
SYMBOL...............................181
SYMBOL AFTER....................108, 182
Syntax error..........12, 84, 200, 264
Sävelvaippa.................43, 127, 145
Sävykkyyssäädin (CONTRAST).......2, 47

TAB..............................82, 170
TAB-näppäin...........................82
TAG..................................182
TAGOFF...............................182
Takaisinkelauspainike (REW)...........65
TAN..................................183
Taukopainike (PAUSE)..................66
Taulukot..............................99
Teksti/ikkunaruudukot................257
TEST.................................183
TESTR................................183
THEN.........................19, 97, 152
Tietokoneen oma kasettinauhuri...5, 64
TIME............................183, 185
TO...............................21, 150
Toistopainike (PLAY)..................65
TRON, TROFF..........................184
TV-vastaanotin.........................3
Tyyppiä osoittavat merkit.............91

UNT..................................184
UPPER$...............................184
USING........................82, 83, 171

VAL..................................184
Ventti-peli..........................102
Vihreä/musta-monitori............1, 47
Vilkkuvat värit.......................30

Virhekoodit ja virheilmoitukset....264
Virtakytkin (POWER)...........2, 3, 4
Voimakkuusvaippa..........41, 126, 147
VPOS.................................185
Vähennyslasku.........................22
Värimonitori.....................2, 46
Värit................25, 109, 247, 251

WAIT.................................185
WEND.................................185
WHILE................................186
WIDTH................................186
WINDOW.....................118, 186, 247
WINDOW SWAP.....................118, 187
WRITE................................187

XOR (POISSULKEVA TAI)...............105
XPOS.................................187

Yhdistelmälaskut.................23, 85
Yhteenlasku...........................21
Ympyrät...............................36
YPOS.................................187

ZONE.............................82, 187

Äänet.......................40, 119, 199
Äänenvoimakkuuden säädin
(VOLUME)........................40, 102
Äänikanavien pito (HOLD)......123, 124
Äänikanavien tahdistus
(RENDEZVOUS)....................122, 123
Äänikanavien tyhjennys (FLUSH).....124
Äänikanavien vapautus ks. RELEASE
Äänityspainike (REC)..................65
Äänivaippojen suunnitteluruudukko..263

AMSTRAD CPC 464:N HARJOITUSKASETIN SISÄLTÖ.
Ohjelmat suomenkielisessä asussa ja testattuina.

MITTARILUKEMA	OHJELMAN NIMI	AIHE	SIVUNUMERO KÄÄNNÖKSESSÄ	SIVUNRO ALKUP.
006 - 008	HARJOITUS 1	INPUT	16	F 2.6
012 - 015	" 2	"	17	F 2.7
016 - 019	" 3	"	17	F 2.7
020 - 023	" 4	KORJAILU	18	F 2.8
028 - 031	" 5	IF - THEN	20	F 2.9
035 - 038	" 6	FOR TO NEXT	21	F 2.10
042 - 045	" 7	" " "	21	F 2.10
048 - 051	" 8	VÄRIEN ESITTELY	31	F 3.7
055 - 058	" 9	LOCATE	33 - 34	F 3.9 - 10
062 - 065	" 10	"	34	F 3.10
069 - 072	" 11	PLOT, DRAW	35	F 3.11
077 - 079	" 12	YMPYRÄ	37	F 3.13
083 - 086	" 13	ORIGIN	38	F 3.14
090 - 093	" 14	"	38	F 3.14
097 - 100	" 15	GOSUB RETURN	39	F 3.15
104 - 107	" 16	YHTEENVETO EDELLISISTÄ	39	F 3.15
111 - 114	" 17	NÄYTTÖTAPA	48	Ch 1.4
118 - 121	" 18	MERKKIVALIKOIMA	57	Ch 1.12
126 - 128	" 19	ZONE, PRINT TAB	82	Ch 3.6
132 - 135	" 20	<, >	87	Ch 4.4
139 - 142	" 21	"	88	Ch 4.4
146 - 149	" 22	"	88	Ch 4.5
151 - 155	" 23	"	89	Ch 4.5
158 - 162	" 24	MUUTTUJAN KOKONAISARVO	95	Ch 4.10
165 - 168	" 25	IF THEN	97 - 98	Ch 4.12
172 - 175	" 26	TAULUKKO	100	Ch 4.14
178 - 182	" 27	DATA, RESTORE	101	Ch 4.15
185 - 188	" 28	DATA, SOUND	101	Ch 4.15
192 - 199	" 29	VENTTIPELI	102 - 104	Ch 4.16
203 - 206	" 30	LOOGISET LAUSEKKEET	106	Ch 4.19
208 - 211	" 31	"	106	Ch 4.19
214 - 218	" 32	"	107	Ch 4.20
221 - 224	" 33	"	107	Ch 4.20
227 - 230	" 34	SKANDINAAVISET MERKIT	108	
233 - 236	" 35	LÄPINÄKYVÄ NÄYTTÖTAPA	110	Ch 5.2
239 - 243	" 36	GRAFIIKAN ESITTELY	112	Ch 5.4
246 - 250	" 37	"	113	Ch 5.5
254 - 257	" 38	SINIKÄYRÄ	113	Ch 5.5
259 - 263	" 39	DRAW, INK	115	Ch 5.7
266 - 274	" 40	3-, 4- TAI 6- SIVUINEN KUVIO	116	Ch 5.8 - 9
277 - 280	" 41	IKKUNAT	118	Ch 5.10